Sommario

4th DIMENSION

4 . LINGUAGGIO

Se più livelli di Break, il Break R0
è riprodotto A FINE DOCUMENTO ma NON
AD OGNI FINE PAGINA (come viceversa con
un unico livello)

- Eseguendo "Esporta Testo", "vol. Not Found"
(Err#-35) non È Trapped da "ON Error"
- Non riesco a creare relazioni NON Automatiche
- Non riesco a inserire il "real" 0.0 in un
campo obbligatorio -

ACI

Prefazione

Nozioni

I comandi

Appendici

Introduzione

4th DIMENSION

4 . LINGUAGGIO

ACI

1. Imparare a programmare

Programmare è un termine molto ampio che indica la creazione di sequenze di istruzioni di ogni genere. Tali istruzioni vanno dalle più semplici alle più complesse, in base alle proprie necessità.
Molto spesso è sufficiente scrivere semplici formule di calcolo: è il caso, ad esempio, di un foglio elettronico.
E' comunque utile automatizzare delle funzioni che con il tempo risulta noioso effettuare manualmente. Create una breve procedura e chiedete di eseguirla. Può succedere che vogliate creare una funzionalità che non è stata predeterminata, ma che è realizzabile grazie al linguaggio estremamente potente di cui è dotato 4th DIMENSION.
Potreste desiderare semplicemente che la vostra applicazione sia molto originale: stabilirete i parametri in base ai vostri gusti e alle vostre necessità.
4th DIMENSION vi permette la massima libertà d'azione. Utilizzatela come meglio preferite.

1. Per programmare

Imparare a programmare non è così complicato come a volte si vuol far credere.
Pur essendo ovvio che non si diventa esperti programmatori dall'oggi al domani, è altresì evidente che programmatori non si nasce. Lo si diventa grazie alla pratica ed all'abitudine.

Se non avete mai utilizzato un linguaggio di programmazione, non lasciatevi intimorire dalle parole.
Quando regolate il vostro videoregistratore per fargli registrare un film mentre non siete in casa, voi state programmando.
Quando utilizzate un foglio elettronico e create una macro, state in realtà programmando. In pratica, ogni qualvolta volete automatizzare una funzione, ogni qualvolta date istruzioni ad una macchina, voi eseguite una programmazione.

Programmare con 4th DIMENSION è un'azione dello stesso tipo. L'unica differenza è che con 4th DIMENSION avete la facoltà di scrivere sia istruzioni estremamente semplici sia sequenze di istruzioni molto complesse. Se siete principianti, accontentatevi, per cominciare, di istruzioni semplici. Di argomento in argomento riuscirete, come molti altri prima di voi, a costruire sequenze molto elaborate.

Non lasciatevi ugualmente intimorire dal vasto numero di comandi che vi propone il linguaggio di 4th DIMENSION. Non c'è nulla che vi obblighi a conoscerli e a gestirli tutti contemporaneamente. Avrete senz' altro occasione di scoprirli poco alla volta, in base alle vostre necessità. E allora, vi renderete conto che è una fortuna che 4th DIMENSION presenti un gran numero di comandi di base: avrete in questo modo un campo d'azione estremamente ampio e diversificato.
Con un linguaggio povero, dotato di pochi comandi, sareste obbligati a cercare combinazioni a non finire per raggiungere i vostri obiettivi, ammesso che sia possibile.
Ciò è valido per i linguaggi di programmazione come per il linguaggio comune: la ricchezza del vocabolario permette d'esprimere, con molte sfumature diverse, una grande quantità di concetti. La mancanza di parole riduce spesso al silenzio o all'impotenza.

2. Dal linguaggio naturale al linguaggio informatico

Un linguaggio di programmazione è un linguaggio come un altro, che, esattamente come l'italiano o l'inglese, possiede un'ortografia, delle regole di sintassi ed una grammatica.
Non vi è che una differenza fondamentale tra un linguaggio naturale ed un linguaggio formale: un linguaggio formale non ammette imprecisioni.

Pronunciando la frase "Mi fai una fotocopia di questo" con tono crescente, chiunque intenderà un imperativo, mentre grammaticalmente avete usato un indicativo.

Un computer non è in grado di interpretare il pensiero.

Il computer ha delle regole di sintassi più rigorose rispetto a quelle usate da un essere umano: non accetterà ambiguità nei vostri comandi.
Ogni comando dato al computer dovrà quindi essere chiaro e rispettare scrupolosamente le regole d'ortografia, di sintassi e di grammatica.
Quando programmate, dovete sempre definire il contesto che rende comprensibile l'istruzione che volete trasmettere.

Imparare a programmare significa abituarsi alla precisione assoluta quando volete comunicare un messaggio.

3. La precisione

Riprendiamo un esempio che abbiamo avuto occasione di realizzare durante la descrizione della creazione di un menu.
Il nostro scopo era quello di personalizzare i menu della nostra applicazione.
Abbiamo quindi creato una procedura di aggiunta schede. La procedura era la seguente:

```
IMPOSTA ARCHIVIO CORRENTE    ([Film])
FORMATO PAGINA   ([Film];"Scheda generale")
Ripeti
  AGGIUNGI SCHEDA
Finché (OK=1)
```

Questa breve procedura comprende tutti gli elementi e i concetti che occorre tenere a mente qualora si desideri scrivere un insieme di istruzioni.

Dove?
Dovete sapere in quale contesto scrivere le vostre istruzioni. Volete agire semplicemente su di un campo, o su di un formato, su di un archivio oppure su tutta la base dati?
Quando nella vita di tutti i giorni parlate senza tener conto del contesto, i casi sono due: o prendete un granchio, oppure non vi si comprende, ed è come se non aveste parlato.
La stessa cosa succede in informatica.

Su che cosa?
Volete che il programma vi permetta di aggiungere delle schede. Ma dove? In quale archivio? Utilizzando quale formato? Dovete dare al programma la possibilità di orientarsi.
Dovete precisare ciò di cui state parlando, su che cosa agite. In altri termini, il programma deve poter identificare sempre chiaramente gli oggetti in questione.
L'archivio su cui si vuole lavorare è l'archivio Film.
Per far sì che il programma identifichi chiaramente che Film è un archivio e non un altro oggetto, questa parola viene posta tra parentesi quadre [Film].
Il nome di un formato è posto tra virgolette: è la convenzione per la scrittura di una costante di tipo alfanumerico.

Non è sufficiente sapere o scrivere le proprie istruzioni, occorre anche strutturarle.
Nella vita corrente, non vi accontentate di mettere insieme le frasi senza coordinarle. Le articolate, le une in rapporto alle altre. Le organizzate. Allo stesso modo, le Parole Chiave predeterminate del programma vi permettono di scrivere delle sequenze coerenti. Le Parole Chiave sono in un certo senso dei controllori logici.
Riprendiamo l'esempio descritto in precedenza.
Potreste scrivere:

```
IMPOSTA ARCHIVIO CORRENTE    ([Film])
FORMATO PAGINA    ([Film];"Completo")
AGGIUNGI SCHEDA
```

Il programma comprenderà ed eseguirà l'azione richiesta.
Ma, in questa procedura, avete dato l'ordine AGGIUNGI SCHEDA, senza tuttavia indicare per quante volte. In mancanza di tale istruzione, il programma proporrà l'aggiunta di una sola scheda alla volta.
Occorre quindi specificare tutto. In particolare, quando volete introdurre varie schede una di seguito all'altra e siete voi a decidere quando terminare l'inserimento. Ne deriva il comando Ripeti ... Finché.

Finché (OK#1)
o meglio ancora, espressione in questo caso del tutto equivalente, dato che OK non può assumere valori diversi da 0 o 1:
Finché (OK=0)

4. Dall'informatica al linguaggio parlato

Se è vero che l'espressione informatica si distingue dal linguaggio naturale per la sua precisione, tuttavia è possibile formulare una certa serie di istruzioni informatiche nel linguaggio naturale. In altri termini, una procedura, benchè complessa, può essere commentata passo passo in modo comprensibile.

Sequenza di comandi

IMPOSTA ARCHIVIO CORRENTE ([Clienti])
CERCA ([Clienti] Città="Torino")
Se (Schede in selezione>0)
 STAMPA SELEZIONE
altrimenti
 AVVISO ("Nessun cliente a Torino")
Fine se

Versione commentata

 ` Prendete l'archivio Clienti
IMPOSTA ARCHIVIO CORRENTE ([Clienti])
 ` Cercate i clienti di Torino
CERCA ([Clienti] Città="Torino")
 ` Se trovate più di 0 schede
Se (Schede in selezione>0)
 ` Stampate le schede
 STAMPA SELEZIONE
altrimenti
 ` Siete informati che non ci sono clienti a Torino
 AVVISO ("Nessun cliente a Torino")
 ` Fine della condizione
Fine se

Quando scrivete una serie di istruzioni, se volete, potete scrivere i vostri commenti. In questo caso fate precedere la riga di commento dall'apostrofo inverso (Opzione-à + Barra spaziatrice).

2. Composizione di questo manuale

La prima parte di questo manuale è costituita da una presentazione generale delle nozioni di base che avete spesso incontrato nei precedenti manuali, all'interno dei brevi esempi di programmazione.

La seconda parte è costituita da una spiegazione di ognuno dei comandi di 4th DIMENSION. Quest'analisi è effettuata per tipologie.

La terza parte è costituita dalle appendici.

Nozioni

4th DIMENSION

4 . LINGUAGGIO

ACI

1. I contesti

Dove scrivere le vostre procedure per renderle operative?
Potete farlo a diversi livelli:
- una serie di istruzioni associata ad un campo o ad una variabile viene denominata **script**.
- una serie di istruzioni associata ad un formato viene denominata **procedura di formato**.
- una serie di istruzioni associata ad un archivio viene denominata **procedura di archivio**.
- una serie di istruzioni associata ad una base dati viene denominata **procedura globale**.

Nota
Abbiamo adottato questa classificazione per una questione di praticità. Letteralmente, una serie di istruzioni indica sempre una serie di istruzioni e non sbagliate dicendo che scrivete una procedura (i due termini sono equivalenti) in uno script. L'unico punto importante è sapere a quale oggetto è associata una procedura, se si tratta di un campo o di una variabile, di un formato, di un archivio o della base dati.

1. Gli script

Uno script rappresenta una serie di istruzioni associata ad un oggetto. Viene eseguito quando l'oggetto a cui è collegato genera un'azione (come succede durante l'inserimento), o quando dev'essere aggiornato (durante la visualizzazione di una selezione, in corso di stampa o d'importazione/esportazione).
Durante l'inserimento, occorre fare clic sul campo o sulla variabile perché lo script venga eseguito. L'attivazione dello script presuppone un'azione diretta da parte dell'utente. Lo script non può essere attivato per mezzo della programmazione.

Esempio di script:
Disponete di un campo Totale netto.
Assegnate uno script a questo campo, e chiedete che l'azione sul campo si ripercuota sulla variabile Totale con IVA.
Scrivete il vostro script a partire dal campo Totale netto:
Totale con IVA:=Totale netto*1,19
Quando confermerete il campo, la variabile verrà aggiornata.

Lo script è molto utile nel caso dei pulsanti, quando, facendo clic, si richiede una azione immediata. E' anche molto utile per i campi la cui modifica comporta un controllo dell'inserimento, o un'azione, come attivare una relazione.

Uno script non è adatto qualora voleste effettuare dei test sull'ambiente di inserimento. Immaginiamo un pulsante di conferma che vorremmo disabilitare fino a quando l'utente non abbia inserito due campi obbligatori.
Potremmo immaginare di associare uno script a ciascuno dei due campi e scrivere:

Se(Campo 1="")|(Campo 2="")
DISABILITA PULSANTE(bCONFERMA)
altrimenti
ABILITA PULSANTE(bCONFERMA)
Fine se

Nota:
La barra verticale è il simbolo di Oppure (Opzione-/)
bCONFERMA è il nome del nostro pulsante.

Dovremmo quindi avere esattamente la stessa serie di istruzioni per due volte. Potete farlo, ma sarebbe una perdita di tempo.
La soluzione migliore consiste nell'avere una procedura associata globalmente al formato. Scriverete la stessa cosa, ma solo una volta.

Gli script in 4th DIMENSION sono molto potenti e facili da utilizzare.

Inoltre, quando copiate e incollate un formato, gli script collegati ai campi ed alle variabili del formato vengono copiati e incollati insieme con il formato, cosa che non accade con le procedure associate ad un formato.
Potete scrivere tutti gli script che desiderate. Fate solo attenzione che gli script non siano contradditori tra di loro. Dato che potete scriverne uno per ogni oggetto, sarà utile non "dimenticare" di avere script disseminati per il formato. La presenza di uno script associato ad un'oggetto è indicata dall'annerimento dell'angolo superiore sinistro dell'oggetto stesso. Avete comunque a disposizione un prezioso aiuto: potete stampare gli script con il comando **Stampa gli Script** nel menu **Archivio**. Questo comando è disponibile quando vi trovate nella finestra per la creazione dei formati.

2. Procedure di formato

Una procedura di formato è associata ad un formato particolare. E' ovvio che, di conseguenza, una procedura di formato non può valere per tutti i formati, a meno di scriverla per ognuno di essi.
Una procedura di formato viene richiamata ad ogni modifica di un qualsiasi oggetto appartenente a quel formato, durante la stampa, la visualizzazione e l'esportazione/importazione dei dati.
Come lo script, non può essere attivata tramite programmazione.
In compenso è richiamata a ogni modifica di qualsiasi oggetto appartenente al formato a cui è associata.
E' quindi particolarmente adatta ai controlli d'ambiente ed in particolare ai controlli d'inserimento, come indicato nell'esempio precedente.

Ovviamente, non è il caso di abusarne.
Ciò che rende interessante una procedura di formato, ne determina anche i limiti:
- viene richiamata costantemente, anche per gli oggetti gestiti dagli script.
- non è specifica di un oggetto.

Immaginate un formato con una cartina d'Europa. Ad ogni nazione è associato un pulsante invisibile, sicché quando fate clic su uno di essi viene visualizzata la scheda informativa relativa a tale nazione.
Potreste scrivere una procedura di formato con cui richiedete a 4th DIMENSION le azioni seguenti.
Quando sono in fase d'inserimento,
se faccio clic sul pulsante invisibile Francia, visualizza la scheda relativa alla Francia ...
se faccio clic sul pulsante invisibile Italia, visualizza la scheda relativa all'Italia ...
se faccio clic sul pulsante invisibile Germania, visualizza la scheda relativa alla Germania ...

```
Se(Durante)
 Caso fra
  :(bFrancia=1)
  CERCA([Nazioni]Nome="Francia")
  MODIFICA SCHEDA([Nazioni])
  :(bGermania=1)
  CERCA([Nazioni]Nome="Germania")
  MODIFICA SCHEDA([Nazioni])
e così via per tutte le altre nazioni d'Europa...
  ...
 Fine caso
Fine se
```

Indipendentemente dal pulsante su cui farete clic, verrà richiamata la procedura di formato completa, e questo per ogni nazione. Potete farlo, ma l'esecuzione sarà più lunga, in quanto 4th DIMENSION dovrà passare in rassegna tutti i casi possibili per trovare il punto in cui avete fatto clic. Uno script è più adatto, in quanto lo script si attiva nel momento stesso in cui fate clic. Scrivete quindi uno script per ogni pulsante invisibile. Lo script per l'Italia sarà il seguente

```
CERCA([Nazioni]Nome="Italia")
MODIFICA SCHEDA([Nazioni])
```

Questo script viene richiamato durante l'inserimento.
Come potete rilevare in base ai due esempi di script e di procedure di formato, avete un certo margine di azione nelle vostre scelte, e ciò che viene messo in gioco in questo lavoro non è tanto o soltanto la tecnica, ma una capacità di analisi da parte vostra che si affinerà con la pratica.

3. Procedure di archivio

Una procedura di archivio è una serie di istruzioni associate ad un archivio.
Viene quindi richiamata ad ogni modifica di un qualsiasi oggetto appartenente all'archivio ed indipendentemente dal formato.
Come per lo script e la procedura di formato, questa non viene attivata per mezzo della programmazione.
A differenza degli script e delle procedure di formato, una procedura di archivio è richiamata soltanto in fase di aggiunta, modifica e importazione di schede.
Ad esempio, non è attivata durante la stampa.

In realtà, è raro aver bisogno di una procedura di archivio: la si può sempre sostituire con una procedura globale richiamata dalle procedure di formato , in quanto di solito non è frequente dover eseguire le stesse operazioni indipendentemente dal formato. In effetti, uno dei casi più adatti all'impiego di una procedura di archivio è nelle operazioni d'importazione, quando è necessario effettuare calcoli sui dati importati.

Se non conoscete 4th DIMENSION ed iniziate con questa versione 4 del programma, potrete fare a meno di questa funzione senza il minimo problema.
Questa possibilità è stata mantenuta per motivi di compatibilità con le versioni precedenti. Vi consigliamo di servirvi delle procedure globali.

4. Procedure globali

Una procedura globale comprende una serie di istruzioni scritte al livello generale della base dati. Ovviamente ciò non significa che debba interessare tutta la base dati in ogni momento, ma che è a disposizione, indipendentemente dalla parte della base dati in cui vi trovate. In altri termini, le procedure sono un insieme di macrocomandi che potete richiamare a piacere per ogni tipo di operazione, nei contesti più disparati.

Riprendiamo l'esempio della cartina dell'Europa.
Abbiamo visto che l'uso degli script permetteva di accelerare l'esecuzione delle nostre istruzioni, eliminando così la fase di ricerca del pulsante su cui è stato fatto clic.
Per essere più precisi, abbiamo quindi assegnato uno script ad ogni pulsante.
Utilizzando una procedura, la nostra applicazione sarà non soltanto più leggibile, ma meglio strutturata.

Prendete un pulsante qualsiasi.
Lo script per il pulsante Italia è il seguente:
CERCA([Nazioni]Nome="Italia")
MODIFICA SCHEDA([Nazioni])

Nota
Utilizziamo il comando MODIFICA SCHEDA nell'ipotesi in cui l'utente volesse apportare delle modifiche.

Lo script per il pulsante Germania:
CERCA([Nazioni]Nome="Germania")
MODIFICA SCHEDA([Nazioni])

Tutti gli script sono identici in ogni loro parte, **ad eccezione** del nome della nazione.
Potete quindi concepire una procedura che gestisca l'identità di struttura: si tratterà di una procedura globale, a cui si passerà un parametro particolare a seconda della nazione.

Si crea quindi una procedura che può essere chiamata *Mostra nazione.*
La procedura è evidentemente la stessa di quella che abbiamo scritto, eccetto la sequenza ([Nazioni]Nome="Nome della nazione").
4th DIMENSION dispone di un potente strumento per designare un parametro il cui contenuto particolare è sconosciuto, ma la cui natura rimane costante: il simbolo del dollaro, $, a cui viene aggiunto un numero: 1,2,3, e così via. In questo caso abbiamo un solo parametro, il nome delle nazioni. Scriviamo $1.
Se teniamo conto anche della sintassi propria del comando CERCA in questo contesto, la procedura globale diventerà:
CERCA([Nazioni]Nome=$1)
MODIFICA SCHEDA([Nazioni])

Nota
Il punto e virgola permette a 4th DIMENSION di distinguere i diversi elementi dei parametri passati. Consultate la sintassi relativa al comando.

A questo punto potrete ritornare ai vostri script e cancellarli tutti, uno dopo l'altro, sostituendoli con:
Mostra nazione("Italia"), per il pulsante Italia, Mostra nazione("Germania") per il pulsante Germania, e così via.

Questo tipo di azione presenta dei notevoli vantaggi:
- la procedura da voi scritta è indicata alla fine della lista dei comandi predeterminati di 4th DIMENSION ed è sufficiente fare clic per utilizzarla. Questa procedura è sempre disponibile;
- vi consente di risparmiare tempo durante l'inserimento (soprattutto quando la procedura sostituisce diverse righe di istruzioni);
- se avete commesso un errore di programmazione, correggerete la procedura una volta soltanto e non in tutte le sue occorrenze;
- infine, mentre gli script, le procedure di formato e le procedure di archivio non possono essere attivate per mezzo della programmazione, le procedure globali possono essere invece richiamate.

5. Conclusione

I contesti nei quali si scrivono le procedure hanno confini ben precisi per quanto riguarda il campo d'azione.
Come descritto nell'esempio di programmazione dei pulsanti per la nostra cartina d'Europa, potete scegliere il contesto a cui associare la procedura: la vostra scelta sarà più o meno vantaggiosa in base all'esperienza che avrete acquisito. Non perdete la pazienza. Mentre imparate a guidare, dovete prima riflettere su ogni azione da compiere, ma, con l'esperienza, queste azioni vi sembreranno naturali.

Per trarre profitto da tutte queste possibilità, vi consigliamo di cercare di comprendere in quale momento verranno eseguite le diverse procedure da voi create ed il loro ordine di esecuzione.
Immaginate ad esempio di avere contemporaneamente degli script, delle procedure di formato e delle procedure globali nella stessa applicazione. Dovete sapere in quale ordine verranno eseguite queste diverse procedure.
E' ciò che chiamiamo **cicli di esecuzione.**

2. Oggetti e sintassi di base

Avete avuto l'occasione di utilizzare i comandi di 4th DIMENSION. Questi comandi agiscono sugli oggetti.
Questo capitolo presenta la lista di questi oggetti e vi spiega l'insieme delle convenzioni di scrittura da rispettare durante il loro uso, convenzioni rappresentate dai simboli per mezzo dei quali 4th DIMENSION li identifica.

1. Archivio

Un archivio è sempre indicato con il proprio nome (massimo 15 caratteri) tra parentesi quadre.
[Ordini]
[Clienti]

2. Formato

Un formato è sempre indicato con il proprio nome tra virgolette (massimo 15 caratteri) o, più generalmente, con un'espressione alfanumerica.
"Inserimento"
"Lista"

Nota per gli sviluppatori
Poiché si tratta di un'espressione alfanumerica, può essere oggetto di calcoli specifici.

3. Campo

L'identificazione di un campo dipende dal contesto in cui scrivete il programma.

- Nelle procedure di archivio, di formato, o negli script dell'archivio al quale appartiene, un campo è identificato con il proprio nome (massimo 15 caratteri):
Totale
Nome

- Negli altri casi, un campo è identificato con il proprio nome preceduto dal nome dell'archivio al quale appartiene:
[Ordini]Totale
[Clienti]Nome

4. Sotto-struttura

L'identificazione di una sotto-struttura (o sotto-archivio) dipende dal contesto in cui scrivete il programma. Sarà identificata:
- con il proprio nome nelle procedure di archivio o di formato relative all'archivio a cui appartiene.
Linee
Indirizzi

- con il proprio nome preceduto dal nome dell'archivio al quale appartiene nelle procedure globali (la convenzione è la stessa di un campo normale)
[Comandi]Linee
[Clienti]Indirizzi

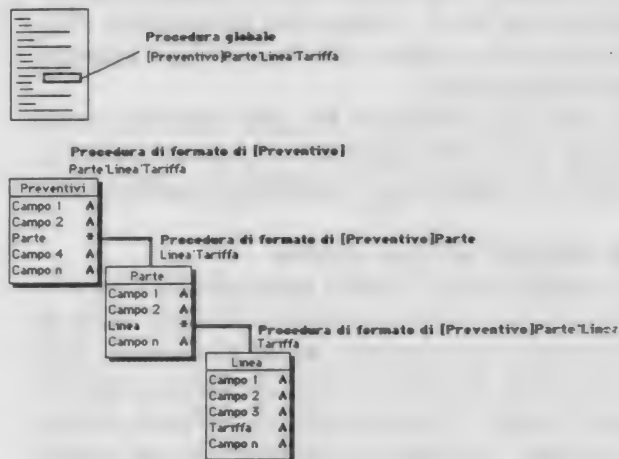
5. Sotto-campo

L'identificazione di un sotto-campo dipende dal contesto in cui scrivete il programma. Sarà indicato:
- con il proprio nome nella procedura di sotto-archivio o di formato del sotto-archivio alla quale appartiene:
Cod_Articolo
CAP

- con il proprio nome preceduto dal nome del sotto-archivio e da un apostrofo nelle procedure di archivio o di formato relative all'archivio a cui appartiene il sotto-archivio.
Linee'Cod_Articolo
Indirizzi'CAP

- con il proprio nome preceduto dal nome dell'archivio, dal nome del sotto-archivio e da un apostrofo nelle procedure globali:
[Comandi]Linee'Cod_Articolo
[Clienti]Indirizzi'CAP

Se avete diversi livelli di sotto-archivi, la sintassi rimane la stessa.
Ogni volta che risalite di un livello, sarà necessario anteporre al nome, come indicato nell' esempio precedente, il nome del sotto-archivio seguito da un apostrofo; in una procedura globale occorrerà indicare il nome completo con il nome dell'archivio.

Esempio
Avete un archivio [Preventivo] che contiene un sotto-archivio Settore, che contiene a sua volta un sotto-archivio Linee, dove Tariffa è un sotto-campo del sotto-archivio Linee.
In base al contesto in cui il sotto-campo Tariffa è usato, verrà identificato in maniera diversa.
Proc. di formato Linee: Tariffa
Proc. di formato Settore: Linee'Tariffa
Proc. di formato [Preventivo]: Settore'Linee'Tariffa
Proc. globale: [Preventivo]Settore'Linee 'Tariffa

6. Variabili

4th DIMENSION presenta due tipi di variabili, indicate con un nome lungo al massimo 11 caratteri: le variabili globali e le variabili locali.
La differenza tra questi due tipi si basa sulla loro accessibilità e sulla durata della loro esistenza.
Una variabile globale è accessibile in qualsiasi contesto della base dati ed esiste fino a quando la base dati è attiva o fino a quando la variabile non viene esplicitamente distrutta.
Una variabile locale è associata ad una procedura, è accessibile soltanto in questa procedura e viene distrutta al termine dell'esecuzione di tale procedura. Tratteremo delle variabili locali al Paragrafo 4 del Capitolo 7 di questa stessa Sezione.

Le variabili globali sono di due tipi:
- le variabili semplici, che hanno un solo valore ben preciso
- le variabili di tipo vettore, composte da più elementi.

Le variabili semplici
Le variabili semplici vengono create nel momento in cui vengono assegnate per la prima volta. 4th DIMENSION le conserva in memoria fino a quando non uscite dall'applicazione o le cancellate. Si tratta quindi di un'informazione temporanea, se paragonata ad un campo, che viene invece registrato in una scheda e che ritrovate ogni volta che riaprite la base dati.
La seconda differenza tra una variabile ed un campo è che una variabile non appartiene ad un archivio e neppure ad una scheda in particolare.

Le variabili di tipo vettore
Le variabili di tipo vettore sono variabili che contengono un numero qualsiasi (limitato dalle dimensioni della memoria RAM) di elementi dello stesso tipo. In effetti, potete dichiarare vettori di reali, di interi, di interi lunghi, di booleani, di testi, di immagini, di date o di puntatori. Queste variabili devono essere dichiarate per mezzo dei comandi VETTORE DI TESTI, VETTORE DI STRINGHE, VETTORE DI REALI, VETTORE DI INTERI, VETTORE DI INTERI LUNGHI, VETTORE DI BOOLEANI, VETTORE DI DATE, VETTORE DI IMMAGINI, VETTORE DI PUNTATORI.

Questo tipo di variabile, molto potente, viene analizzata nella sezione di questo manuale relativo ai comandi per la gestione dei vettori.

7. Insieme

4th DIMENSION vi permette di gestire oggetti particolari, chiamati insiemi, che vengono creati per mezzo dei comandi per la gestione degli insiemi (consultate il relativo capitolo in questo stesso manuale).
Un insieme è sempre indicato con il proprio nome tra virgolette (massimo 80 caratteri) o, più generalmente, con un'espressione alfanumerica il cui valore è identico al suo nome.
"Schede da cancellare"
"Ordini del cliente"

8. Costante

Una costante è un valore che rimane invariato.
Una costante può essere di quattro tipi: alfanumerico, numerico, data e ora.
Non si può agire su una costante. E' un oggetto fisso che si scrive nel programma o, in altri termini, che fa parte del "codice" della vostra applicazione.

Costanti di tipo alfanumerico
Una costante alfanumerica si scrive sempre racchiusa tra virgolette:
Nome:="Italia"
In questo esempio, la costante è "Italia".

Costanti di tipo numerico
Le costanti numeriche si scrivono senza segni di separazione:
Totale:=586
In questo esempio, la costante è 586.

Costanti di tipo data
Le costanti di tipo data devono essere inserite sempre fra due punti esclamativi:
Anniversario:=!1/1/89!
In questo esempio, la costante è !1/1/89!.

Costanti di tipo ora

Le costanti del tipo ora devono essere sempre inserite fra i simboli † (Opzione-t):
Ora arrivo:=†1:45:32†
In questo esempio la costante è †1:45:32†

9. Le espressioni

Un'espressione è una formula composta da elementi collegati fra loro da operatori.
3+4 è un'espressione il cui valore è 7. In questo esempio, + è un operatore che effettua l'addizione delle due espressioni primarie 3 e 4.

Una espressione può essere composta dai seguenti elementi (a destra del simbolo ":="):

- una costante
a:=3
b:="Buongiorno"
c:=!14/7/89!

- un campo
a:=[Contatti]Nome

- una variabile
b:=Var1

- una funzione
d:=**Data odierna**

Durante l'esecuzione del programma l'espressione sarà valutata (cioé calcolata) e fornirà un valore di un certo tipo. In realtà, il fatto di scrivere 3+4 darà il valore 7 di tipo numerico, che è diverso da "3"+"4" che darà un valore alfanumerico uguale a "34".

a:=(3*Var1)+(2*Var2)
è un'espressione di tipo numerico se Var1 e Var2 sono variabili di tipo numerico.

d:=**Data odierna**+5
è un'espressione di tipo data.

h:=**Ora attuale**
è un'espressione di tipo ora.

b:=(OK=1)
è un'espressione di tipo booleano.

Al momento della valutazione di un'espressione, 4th DIMENSION non fa differenza fra un *tipo alfanumerico* ed un *tipo testo*, come nel caso dei campi. Un'espressione è semplicemente di tipo testo qualora si tratti di una stringa di caratteri. Allo stesso modo, un'espressione è solo di tipo *numerico*, senza distinzione fra tipi numerici.
I tipi di espressione sono quindi 6: *testo, numerico, data, ora, booleano* e immagine.

3. Gli operatori

Un operatore è un simbolo che permette di effettuare un calcolo tra due valori.
Quindi, il simbolo * corrisponde ad una moltiplicazione di due valori numerici.

4th DIMENSION possiede operatori sui valori numerici, ma anche sulle stringhe di caratteri, sulle date, sui booleani e persino sulle immagini.
Gli operatori sui valori numerici vi sembreranno, ovviamente, facili da usare, ma vi consigliamo comunque di leggere il capitolo seguente.
Leggete attentamente gli operatori relativi agli altri tipi di valori: vi permetteranno di effettuare operazioni che non vi sareste mai aspettati di poter eseguire.

Nota per i programmatori in linguaggi tipo Pascal
In 4th DIMENSION, non esiste il concetto di precedenza degli operatori. Un'espressione viene calcolata partendo da sinistra verso destra.

1. Operatori sui valori numerici

Addizione

Simbolo +

L'operatore + vi permette di effettuare la somma di 2 espressioni di tipo numerico.

Num1 + Num2

[Clienti]Fatturato + [Fatture]Totale
fornisce il fatturato del singolo cliente + il totale della fattura.
Vale a dire: 5 + 65 restituisce 70

Sottrazione

Simbolo -

L'operatore - vi permette di effettuare una sottrazione fra due espressioni di tipo numerico.

Num1 - Num2

[Clienti]Fatturato - [Avere]Totale
fornisce il fatturato del cliente - l'importo dell'avere
Vale a dire: 180 - 5 restituisce 175

Moltiplicazione

Simbolo *

L'operatore * vi permette di effettuare la moltiplicazione di due espressioni di tipo numerico.

Num1 * Num2

[Fatture]Importo netto*0,19
fornisce l'importo dell'IVA al 19%
Vale a dire: 5*10 restituisce 50

Divisione

Simbolo /

L'operatore / vi permette di dividere una espressione di tipo numerico per un'altra.

Num1 / Num2

Vale a dire: 20 / 5 restituisce 4

Divisione intera

Simbolo \

L'operatore \ vi permette di dividere una espressione di tipo numerico per un'altra restituendo il risultato intero.

Num1 \ Num2

Vale a dire: 10\3 restituisce 3

Modulo

Simbolo %

L'operatore % fornisce il resto della divisione intera fra due espressioni di tipo numerico
Vale a dire: 10%3, restituisce 1
10%2, restituisce 0

Elevamento a potenza

Simbolo ^

L'operatore ^ vi permette di elevare *Num1* alla potenza *Num2* (^ si ottiene scrivendo il carattere ^ e, successivamente, la barra spaziatrice).

Num1 ^ Num2

Vale a dire:
2^3 restituisce 8 e si legge 2 alla terza
2^0,5 restituisce 1,414 e si legge radice quadrata di 2.

2. Operatori sui valori di tipo data

Differenza

Simbolo -

L'operatore - utilizzato tra due date vi restituisce il numero di giorni che separano queste due date.

Data1 - Data2

Nota
Questo operatore utilizza le routines del System di Macintosh che gestiscono unicamente le date a partire dal 1° gennaio 1904.

[Fatture]Pagamento-[Fatture]DataFattura
restituisce il termine di pagamento di una fattura.

Data odierna - !1/1/89!
restituisce il numero di giorni trascorsi dall'inizio del 1989
(Come già visto sopra, una costante di tipo data viene inserita fra due punti esclamativi).

Differenza tra una data ed un'espressione numerica

L'operatore - può essere utilizzato tra una data ed un'espressione numerica.

Data1 - Espr_Num

restituisce quindi una data uguale a *Data1* meno un numero di giorni pari a *Espr_Num*.

Data del giorno - 1
restituisce la data di ieri.

Addizione
Simbolo +
L'operatore + può essere utilizzato tra una data ed un'espressione numerica.
Data1 + Espr_Num
restituisce quindi una data uguale a *Data1* più un numero di giorni pari a *Espr_Num*.

[Fatture]DataFattura + 30
restituisce una data corrispondente alla data della fattura + 30 giorni

Data del giorno + 1
restituisce la data di domani.

3. Operatori sui valori di tipo stringa

Concatenazione
Simbolo +
Concatenare significa "mettere l'uno dopo l'altro".
L'operatore + utilizzato tra due espressioni alfanumeriche vi restituisce una stringa di caratteri composta da *Stringa1* seguita dai caratteri che compongono la *Stringa2*
Stringa1 + Stringa2

"Buongiorno "+"Signora"
restituisce "Buongiorno Signora"

Ripetizione
Simbolo *
L'operatore * utilizzato tra un valore alfanumerico ed un valore numerico ha come risultato una stringa di caratteri composta da *Stringa1* ripetuta per un numero di volte corrispondente a *Num1*
Stringa1 * Num1

"****"*10
restituisce "********************"

Nota
La combinazione di tasti per ottenere il carattere " "dipende dal tipo di carattere in uso.

≤Indice≥
I segni ≤ e ≥ si ottengono premendo Opzione–< oppure Opzione->.
Questi due operatori vi permettono l'accesso all'ennesimo carattere di una stringa.

Prendiamo la stringa seguente:
v:="signora"

se scrivete v≤1≥:="S"
v assume il valore di "Signora"

Nota
Non potete cambiare la lunghezza di una stringa di caratteri con questo metodo. Se l'indice da voi utilizzato è superiore alla lunghezza della vostra stringa, il comando verrà ignorato.

4. Operatori sui valori di tipo boleano

Congiunzione
Simbolo &
L'operatore & restituisce un valore booleano Vero se le due espressioni *Booleano1* e *Booleano2* sono entrambe Vere. In tutti gli altri casi, darà valore Falso.
Questo operatore viene anche chiamato "E logico".
Booleano1 & Booleano2

Se (OK=1) & (Schede in selezione=1)
Il test avrà come risultato Vero se le due condizioni sono entrambe Vere.

Disgiunzione
Simbolo |
L'operatore | (ottenuto premendo Opzione-:) restituisce un valore booleano Vero se una delle due espressioni *Booleano1* e *Booleano2* è Vera.
Restituisce un valore Falso soltanto se entrambe le espressioni non sono verificate.
Questo operatore viene anche chiamato "O Logico".
Booleano1 | Booleano2

5. Operatori di confronto

Gli operatori di confronto permettono di confrontare fra loro gli oggetti gestiti in 4th DIMENSION.
Questi operatori possono essere applicati a tutti i tipi di oggetto, ad eccezione delle immagini e dei vettori.
Nel caso dei puntatori, sono utilizzabili soltanto l'uguaglianza e la disuguaglianza.

(Per il concetto di puntatore, fate riferimento alla sezione relativa ai comandi).

Le stringhe di caratteri vengono confrontate sulla base del codice ASCII dei caratteri stessi. Pertanto, *A* è inferiore a *B*, in quanto il codice ASCII di *A* è 65, mentre quello di *B* è 66.
D'altra parte, i confronti fra stringhe non tengono conto delle maiuscole, delle minuscole e degli accenti.
Quindi, *Signore* è uguale a *siGNORE* e *àncora* è uguale ad *ancora*.

Uguaglianza
Simbolo =
L'operatore = restituisce un valore Vero se le due espressioni sono identiche.
Espressione1 = Espressione2

"Buongiorno"="Signora"
restituisce Falso.

Data del giorno = !1/1/89!
restituisce Falso, eccetto per l'1 gennaio 1989.

4587=458
restituisce Falso.

Diversità
Simbolo #
L'operatore # restituisce un valore Vero se le due espressioni sono diverse.
Espressione1 # Espressione2

"Buongiorno"#"Signora"
restituisce Vero.

Data del giorno # !1/1/89!
restituisce Vero, eccetto per l'1 gennaio 1989

4587 # 458
restituisce Vero.

Maggioranza
Simbolo >
L'operatore > restituisce un valore Vero se *Epressione1* è maggiore di *Espressione2*.
Si tratta di una maggioranza stretta. Se le due espressioni sono uguali, il valore dato sarà Falso.
Espressione1 > Espressione2

"Signore">"Signora"
restituisce Vero, non per maschilismo, ma perchè l'ultima *e* ha un codice ASCII superiore a quello della *a*.

!14/7/89!>Data odierna
restituisce Falso fino al 14 luglio 1989.

Maggioranza non stretta
Simbolo >=
L'operatore >= restituisce un valore Vero se *Espressione1* è maggiore o uguale a *Espressione2.*
Espressione1 >= Espressione2

Minoranza
Simbolo <
L'operatore < restituisce un valore Vero se *Espressio- ne1* è minore di *Espressione2*.
Si tratta di una minoranza stretta.
Se le due espressioni sono uguali, il valore dato sarà Falso.
Espressione1 < Espressione2.

Minoranza non stretta
Simbolo <=
L'operatore <= restituisce un valore Vero se *Espressione1* è minore o uguale a *Espressione2*.
Espressione1 <= Espressione2

6. Operatori sulle immagini

Concatenazione orizzontale
Simbolo +
La concatenazione orizzontale permette di collegare immagini in senso orizzontale.
Sintassi: Immagine1 + Immagine2

Parametri: Immagine1 Espr_Immagine
Immagine2 Espr_Immagine

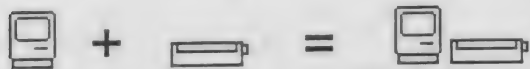



Immagine2 viene collocata a destra di *Immagine1*. Le due immagini sono allineate rispetto alla base.

Attenzione
Immagine1 + *Immagine2* non è equivalente a *Immagine2* + *Immagine1*.
La concatenazione orizzontale non è un'operatore commutativo.

Concatenazione verticale
Simbolo /
Sintassi: Immagine1 / Immagine2
Parametri: Immagine1 Espr_Immagine
Immagine2 Espr_Immagine

Immagine1 viene collocata sopra a *Immagine2*.
Le due immagini sono allineate rispetto al proprio lato sinistro.

Attenzione
Immagine1/Immagine2 non è l'equivalente di *Immagine2/Immagine1*.
La concatenazione verticale non è un operatore commutativo.

Sovrapposizione esclusiva
Simbolo &
Sintassi: Immagine1 & Immagine2
Parametri: Immagine1 Espr_Immagine
Immagine2 Espr_Immagine

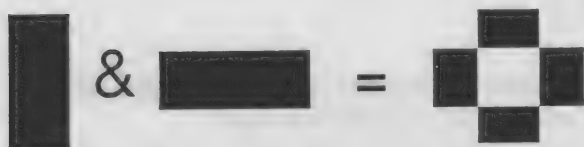



Immagine1 e *Immagine2* vengono sovrapposte ed allineate rispetto al proprio centro.
Ciascun punto dell'espressione immagine che ne deriva sarà nero se uno dei punti corrispondenti di *Immagine1* o di *Immagine2* è nero, ma non lo sarà se lo sono entrambi.

Sovrapposizione inclusiva
Simbolo |
Sintassi: Immagine1 | Immagine2
Parametri: Immagine1 Espr_Immagine
Immagine2 Espr_Immagine

Immagine1 e *Immagine2* vengono sovrapposte ed allineate rispetto al proprio centro.
Ogni punto dell'espressione immagine che ne deriva sarà nero se uno o entrambi i punti corrispondenti di *Immagine1* o *Immagine2* è nero.

Spostamento orizzontale
Simbolo +
Sintassi: Immagine1 + Num1
Parametri: Immagine1 Espr_Immagine
Num1 Espr_Num

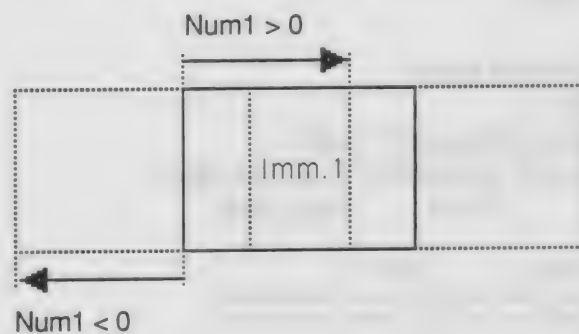
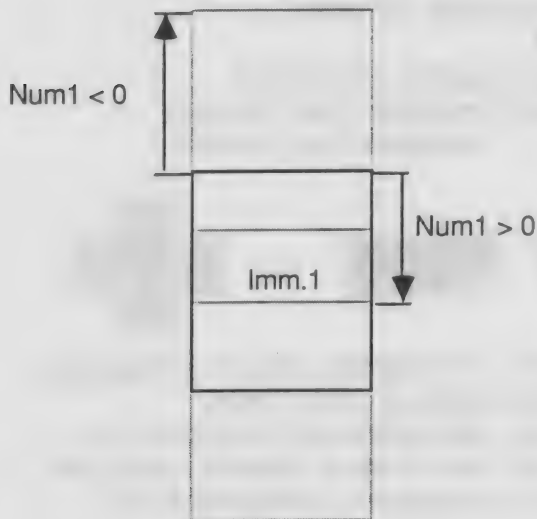
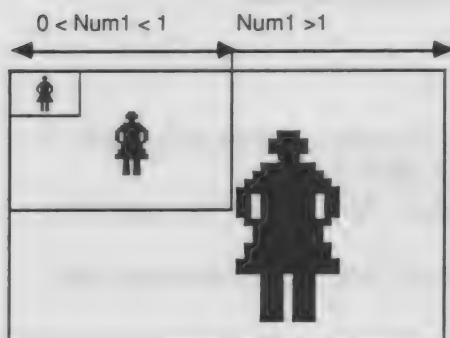
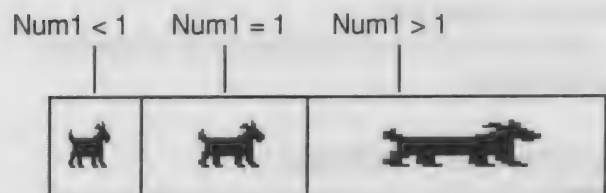
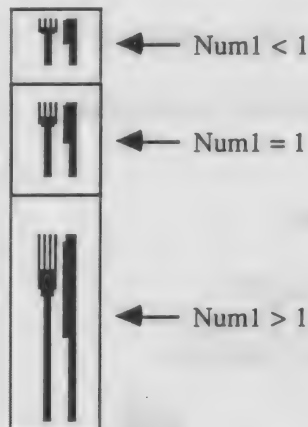



Immagine1 viene spostata in senso orizzontale di un numero di punti pari a *Num1*.
Se *Num1* è positivo, l'immagine verrà spostata verso destra.
Se *Num1* è negativo, l'immmagine verrà spostata verso sinistra.

Spostamento verticale
Simbolo /
Sintassi: Immagine1 / Num1
Parametri: Immagine1 Espr_Immagine
Num1 Espr_Num

L'immagine *Immagine1* viene spostata in senso verticale di un numero di punti pari a *Num1*.
Se *Num1* è positivo, l'immagine verrà spostata verso il basso.
Se *Num1* è negativo, l'immagine verrà spostata verso l'alto.

Riduzione in scala
Simbolo *
Sintassi: Immagine1 * Num1
Parametri: Immagine1 Espr_Immagine
Num1 Espr_Num

L'espressione *Immagine1* cambia le proprie dimensioni in funzione del coefficiente *Num1*.
Se *Num1* è compreso fra 0 e 1, l'immagine verrà ridotta. Se *Num1* è superiore a 1, l'immagine verrà ingrandita.

Estensione orizzontale
Simbolo *+
Sintassi: Immagine1 *+ Num1
Parametri: Immagine1 Espr_Immagine
Num1 Espr_Num

L'espressione *Immagine1* cambia la propria larghezza in funzione del coefficiente *Num1*.
Se *Num1* è superiore a 1, l'immagine sarà ristretta.
Se *Num1* è inferiore a 1, l'immagine sarà allargata.

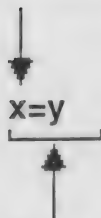
Estensione verticale
Simbolo */
Sintassi: Immagine1 */ Num1
Parametri: Immagine1 Espr_Immagine
Num1 Espr_Num

L'espressione *Immagine1* cambia la propria altezza in funzione del coefficiente *Num1*.
Se *Num1* è inferiore a 1, l'immagine verrà accorciata. Se *Num1* è superiore a 1, l'immagine verrà allungata.

4. Uguaglianza e assegnamento

Avete avuto diverse volte l'occasione di utilizzare il simbolo di uguaglianza = e quello di assegnamento :=
Questo paragrafo ha la sola funzione di ricordarvi la differenza principale tra questi due concetti.

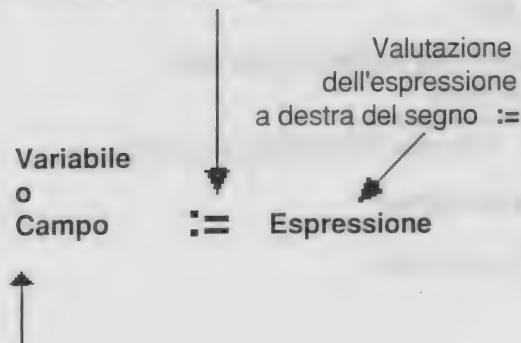
1. Il test d'uguaglianza =

Il simbolo = è un simbolo di confronto
Il test d'uguaglianza restituisce un valore Vero o Falso.

Simbolo d'uguaglianza

Questo insieme costituisce un'espressione logica che restituisce un valore Vero o Falso.

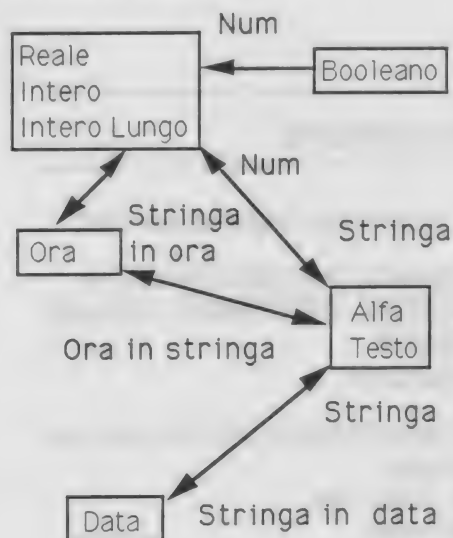
2. L'assegnamento :=

Il simbolo := viene utilizzato come simbolo di assegnamento.
Un' assegnamento consiste nel ricopiare un valore o il risultato di un calcolo in un altro valore.
Viene pertanto scomposto nel modo seguente:

Simbolo di assegnamento

Il risultato dell'espressione viene ricopiato nella variabile o nel campo.

Esempio
AVar:=**Schede in archivio**([Fatture])/10

L'esecuzione di questa istruzione prenderà in considerazione innanzitutto la parte destra, dividendo il numero di schede dell'archivio ([Fatture]) per 10, poi ne ricopierà il risultato nella variabile AVar.

Assegnamento dei valori ai campi
Dovete assegnare un valore corrispondente al tipo di campo.
Se avete un campo Nome, di tipo alfanumerico, non potrete scrivere:
Nome:=10
poichè 10 in questo caso è una costante numerica e non una costante alfanumerica.
Potete ovviamente effettuare delle conversioni;
Eccone lo schema:

In neretto sono indicate le funzioni di conversione. Ognuna di esse è descritta in dettaglio nella sezione che riguarda i comandi.

Assegnamento dei valori alle variabili
L' assegnamento ad una variabile ne determina il tipo.
Se scrivete v:=10, v diventa una variabile numerica.
Se, più avanti, scriverete v:="Buongiorno", questa variabile diventerà alfanumerica.
In 4th DIMENSION tale cambiamento di tipo è possibile; comunque, è consigliabile evitare questa operazione se avete intenzione di compilare la vostra applicazione.
A questo proposito, consultate la seconda parte di questo manuale.

L'assegnamento di una funzione ad una variabile ne determina il tipo.

Note
- Se una variabile compare in un formato senza essere mai stata assegnata, questa sarà di tipo alfanumerico.
- La formattazione di una variabile in un formato non costituisce di per sé una definizione del tipo della variabile.

5. Le parole chiave

Quando scrivete le istruzioni esistono due casi:

- scrivete una sequenza di azioni concatenate
IMPOSTA ARCHIVIO CORRENTE ([Clienti])
TUTTE LE SCHEDE
STAMPA SELEZIONE

- scrivete sequenze articolate tra di loro per mezzo delle parole chiave:
Se...altrimenti...Fine se
Caso fra...Fine caso
Mentre...Fine mentre
Ciclo...Fine ciclo
Ripeti...Finché

Queste Parole Chiave sono in un certo senso dei controllori della sequenza delle operazioni.
Se...altrimenti...Fine se e **Caso fra...Fine caso** sono test logici.

In questo caso potrete scegliere fra diverse alternative in base al risultato di un test condizionale.

Mentre...Fine mentre, Ciclo...Fine ciclo, Ripeti...Finché sono cicli. Nel caso dei cicli, specificherete un insieme di comandi, definendo le condizioni ed i limiti della loro ripetizione.

1. Se...altrimenti...Fine se
(If...Else...Endif)

Il Se permette di scegliere quale sequenza eseguire in funzione di una condizione Vera o Falsa.
Questa condizione è sempre espressa tra parentesi.
Se(...)
Il **Se** è lo strumento principale: permette di indirizzare l'esecuzione della procedura in una maniera o in un'altra.
Scriverete quindi:

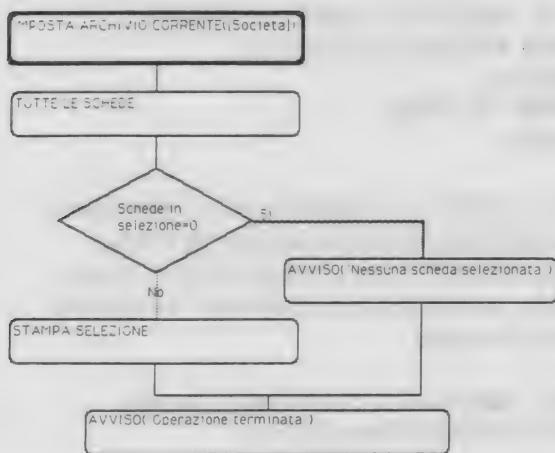
```
Se (Espr_Logica)
   Sequenza1
altrimenti
   Sequenza2
Fine se
```

La seconda sequenza (quella preceduta da **altrimenti**) non è sempre obbligatoria. Può succedere che abbiate un test che esegue una sequenza soltanto se la condizione è Vera. In questo caso scriverete:

```
Se (Espr_Logica)
   Sequenza1
Fine se
```

Esempi
Versione a lista:

```
IMPOSTA ARCHIVIO CORRENTE ([Società])
TUTTE LE SCHEDE
Se (Schede in selezione =0)
   AVVISO ("Non esistono schede selezionate")
altrimenti
   STAMPA SELEZIONE
Fine se
AVVISO ("Ho terminato")
```

Versione a blocchi

Potete inoltre collocare più test su diversi livelli di annidamento:

```
IMPOSTA ARCHIVIO CORRENTE ([Clienti])
CERCA
Se (OK=1)
    ORDINA
    Se (OK=1)
        CONFERMA ("Vuoi stampare?")
        Se (OK=1)
            STAMPA SELEZIONE
        Fine se
    Fine se
Fine se
```

Nota

Se osservate i risultati dei sondaggi d'opinione, a domande del tipo: "Vi piacciono le fragole?", esiste sempre una percentuale di popolazione che non ha idee in merito e che risponde "Non so".
Un computer non è mai indeciso, ad una domanda risponderà sempre con un Sì o con un No.
Una condizione logica sarà sempre Vera o Falsa, mai indeterminata.

2. Caso fra...Fine caso
(Case of...End case)

Gli annidamenti di **Se...altrimenti...Fine se** possono diventare fastidiosi da gestire se i casi da verificare sono molti.
La scansione dei casi diventa più chiara con l'uso di **Caso fra...**
La sintassi è la seguente: i casi in esame sono preceduti dal segno **:**

```
Caso fra
 :([Bambini]Età<3)
 [Bambini]Regalo:="Peluche"
 :([Bambini]Età<8)
 [Bambini]Regalo:="Lego"
 :([Bambini]Età<12)
 [Bambini]Regalo:="Meccano"
 :([Bambini]Età<15)
 [Bambini]Regalo:="Libro"
altrimenti
 AVVISO("Nessun regalo")
Fine se
```

Potreste scrivere lo stesso insieme di test usando **Se...Fine se.**
In questo caso, la procedura sarebbe la seguente:

```
Se([Bambini]Età<3)
 [Bambini]Regalo:="Peluche"
altrimenti
 Se([Bambini]Età<8)
  [Bambini]Regalo:="Lego"
 altrimenti
  Se([Bambini]Età<12)
   [Bambini]Regalo:="Meccano"
  altrimenti
   Se([Bambini]Età<15
    [Bambini]Regalo:="Libro"
   altrimenti
    AVVISO("Nessun regalo")
   Fine se
  Fine se
 Fine se
Fine se
```

La seconda differenza tra il test **Se...Fine se** e **Caso fra...Fine caso** è la seguente: quando 4th DIMENSION incontra la prima condizione Vera, esegue la relativa sequenza, senza verificare gli altri casi, quindi esce da **Caso fra**.
Se nessuna delle condizioni di **Caso fra** è vera, 4th DIMENSION esegue la sequenza che si trova nella parte **altrimenti**, se questa è stata prevista. Se nella parte **altrimenti** del **Caso fra** non esiste nessuna sequenza, non succede nulla, e 4th DIMENSION riprende l'esecuzione dopo il **Fine caso**.

Note:
- Per chi programma in Pascal, il funzionamento di "Caso fra" è diverso dal "Case of" Pascal. In linguaggio Pascal, le condizioni rappresentano diversi valori dello stesso oggetto, nel "Caso fra" di 4th DIMENSION le condizioni non hanno alcun legame tra di loro, ma si tratta semplicemente di una sequenza di espressioni logiche.
- L'esecuzione di un "Caso fra" è più veloce rispetto ad una serie di test annidati.

3. Mentre...Fine mentre
(While...Endwhile)

Per rappresentare l'azione di **Mentre**, immaginate di essere un magazziniere e di dover prendere un pacco che si trova in magazzino; per prendere il pacco, dovete essere all'altezza giusta.
Andate a prendere una scala. Salite un gradino e guardate se siete all'altezza giusta. Se non lo siete, continuate a salire fino a quando non sarete in grado di prendere il pacco.
Se il pacco fosse stato alla vostra portata sin dall'inizio, non avreste avuto bisogno di salire sulla scala.

Mentre non sono all'altezza giusta
Salgo un gradino
Fine mentre
Prendo il pacco.

Il ciclo **Mentre** è un ciclo condizionale, ovvero viene eseguito fintantoché la condizione testata è vera. E' quindi utile per ripetere un'operazione per un numero di volte non determinato in anticipo, numero che dipende dall'esecuzione del ciclo stesso.

Il principio di scrittura è semplice:

```
Mentre(Condizione)
    ...
Fine mentre
```

Il ciclo **Mentre** viene eseguito fintantoché l'espressione *Condizione* è vera.
E' importante notare che il test di *Condizione* viene effettuato prima dell'ingresso nel ciclo.
Quindi se *Condizione* è falsa, non si entra inutilmente nel ciclo.

Esempio:

```
IMPOSTA ARCHIVIO CORRENTE([Clienti])
FORMATO PAGINA("Inserimento")
Mentre(OK=1)
 AGGIUNGI SCHEDA
Fine mentre
```

Se OK=0, questa procedura non aggiungerà nessuna scheda, perché la condizione è falsa.
Per essere sicuri di poter passare almeno una volta nel ciclo, consigliamo di impostare la condizione a vero scrivendo:

```
IMPOSTA ARCHIVIO CORRENTE([Clienti])
FORMATO PAGINA("Inserimento")
OK:=1
Mentre(OK=1)
 AGGIUNGI SCHEDA
Fine mentre
```

Occorre anche fare attenzione al fatto che la condizione posta deve essere modificata nel ciclo, altrimenti il programma eseguirà un ciclo senza fine.

4. Ciclo...Fine ciclo
(For...Endfor)

Il principio del ciclo viene utilizzato molto spesso nella vita quotidiana.
Ad esempio, quando dite ad un fattorino:

1. "Vada al magazzino"
2. "Mi scarichi 50 casse dal camion"
3. "Ritorni quando avrà finito"

Avete chiesto all'addetto di ripetere 50 volte l'operazione di scarico di una cassa.
Effettivamente è più semplice dire "Scarichi 50 casse ...", anziché "Scarichi, Scarichi, ..." 50 volte di seguito.
Potete esprimere la stessa cosa nel linguaggio informatico, chiedendo al computer di ripetere le stesse operazioni 10, 50, 100 o un milione di volte.

```
`Esegui un ciclo contando da 1 a 100
Ciclo (i;1;100)
`Fine della sequenza, ritorno alla prima riga
Fine ciclo
```

In questo esempio, i è una variabile incrementata automaticamente da 4th DIMENSION.

L'istruzione **Ciclo** permette quindi di eseguire un ciclo per un numero predeterminato di volte.
Si rivela quindi molto utile qualora desideriate effettuare una sequenza per un numero di volte noto in anticipo.
Se volete passare in rassegna tutte le schede di un archivio, conoscendo questo numero in anticipo, potrete utilizzare l'istruzione **Ciclo**.
La sintassi è la seguente:

Ciclo(Contatore; Inizio; Fine; {Passo})
 Sequenza di istruzioni
Fine ciclo

Contatore è una variabile che serve a contare il numero di passaggi nel ciclo.
Inizio fornisce il valore di partenza di *Contatore*
Fine fornisce il valore di arresto di *Contatore*
Passo è opzionale e permette di stabilire l'incremento di *Contatore* ad ogni ciclo.

All'avvio di questo Ciclo, la variabile *Contatore* assumerà il valore *Inizio*.
Contatore sarà incrementata automaticamente del valore di Passo (pari ad 1 se Passo non è specificato) fino al raggiungimento del valore *Fine*.

Note:
- Contatore è una variabile numerica.
- L'istruzione Ciclo è più rapida rispetto alle altre strutture del ciclo (Mentre, Ripeti), in quanto 4th DIMENSION incrementa automaticamente la variabile che serve da contatore.
- Passo deve essere un valore numerico positivo.

Esempio
Volete far comparire il numero di fatture emesse ad ogni mese.
Dovrete creare un ciclo sul numero dei mesi, cioé 12, in cui chiedete di far comparire una finestra di avviso, che mostri il numero di schede trovate.

```
Ciclo(vmese;1;12)
 CERCA([Fatture]Mese=vmese)
 AVVISO("Fatture:"+Stringa(Schede in selezione))
Fine ciclo
```

5. Ripeti...Finché
(Repeat...Until)

Il ciclo **Ripeti** è condizionale, come il ciclo **Mentre**.
La differenza è che la condizione viene posta alla fine del ciclo.E' quindi utile per ripetere un'operazione per un numero di volte non prestabilito, che dipende dall'esecuzione del ciclo stesso.

La struttura generale di questa istruzione è di questo tipo:

Ripeti
 ...
Finché(Condizione)

Il ciclo **Ripeti** viene eseguito finché l'espressione *Condizione* non diventerà vera.
E' importante notare che il test di *Condizione* viene effettuato alla fine del primo passaggio, che verrà quindi eseguito almeno una volta.
Notate che, contrariamente a **Mentre**, **Ripeti** viene eseguito finché l'espressione *Condizione* non sarà vera. La scrittura dell'esempio precedente, utilizzando **Ripeti**, avviene in questo modo:

```
IMPOSTA ARCHIVIO CORRENTE([Clienti])
FORMATO PAGINA("Inserimento")
Ripeti
 AGGIUNGI SCHEDA
Finché(OK=0)
```

6. Cicli di esecuzione delle procedure di formato

Nel momento in cui definite le istruzioni, cercate di capire quando verranno eseguite.
In effetti, esistono 3 fasi:

Prima
Durante
Dopo

Queste fasi permettono di distribuire le azioni da effettuare, pertanto sono sempre associate a operatori che consentono delle scelte in base alle condizioni. In particolare: **Se...** e **Caso fra ...**

Nota
Prima, Durante, Dopo *hanno un valore più generale del semplice riferimento all'inserimento di schede. L'inserimento è inteso come un'azione qualunque sulla base dati. Alla lettera, si dovrebbe soltanto dire* ***Prima, Durante, Dopo*** *un'azione in generale.*
Esistono inoltre altre 3 fasi più specifiche richiamate al momento della visualizzazione e della stampa. Si tratta di:
In intestazione
In rottura
In pié di pagina

L'utilizzo di queste fasi è definito nello schema generale di esecuzione dei cicli, allegato a questa sezione.

1. Prima

Questa fase è richiamata una sola volta, prima che la scheda appaia sullo schermo.
Consente di effettuare un calcolo prima che il formato appaia sullo schermo.
Questo ciclo è quindi molto utile per definire valori standard, o per preparare un inserimento aggiornando certe informazioni.

2. Durante

Questa fase viene richiamata ogni volta che si modifica un'informazione (modifica di un campo, clic su un pulsante, su un pulsante d'opzione, su un radio bottone, apertura di un menu...).
E' il punto centrale delle procedure durante l'inserimento.
Vi serve a controllare l'inserimento, a fare calcoli su una fatturazione, a mettere un nome in maiuscolo, e così via.

Nota:
Nella finestra di dialogo per la definizione delle variabili in ambiente Progetto, il pulsante d'opzione **Esegui solo se "Durante"** è attivato. Se desiderate l'esecuzione dello script collegato alla variabile in altre fasi, non dimenticate di disattivare l'opzione.

3. Dopo

Questa fase viene richiamata una sola volta e soltanto se la scheda è stata confermata.
Questa fase serve quindi ad effettuare i calcoli che desiderate fare solo se la scheda è registrata, come ad esempio l'aggiornamento di un conto clienti in una fatturazione.
Naturalmente, è possibile modificare il valore di qualsiasi campo dell'archivio nella fase Dopo.

7. Strumenti di programmazione

Se siete principianti, questo capitolo vi richiederà maggiore attenzione, in particolare i Paragrafi 2, 3 e 4. Vi permetterà di guadagnare tempo nella strutturazione del vostro lavoro, anche se doveste scrivere soltanto formule semplici.

1. I due tipi di comandi

Quando programmate in 4th DIMENSION, vi vengono proposti due tipi di comandi:
Comandi in minuscolo: si tratta di **funzioni**.
Comandi in maiuscolo: si tratta dei **comandi** veri e propri.

Una **funzione** si differenzia da un comando perché la funzione restituisce un valore, mentre il comando effettua un'azione senza restituire valori di ritorno.
TUTTE LE SCHEDE è un comando che seleziona tutte le schede di un archivio. Questo comando non restituisce alcun valore di ritorno.
Schede in selezione è una funzione che fornisce il numero di schede selezionate in un archivio.

2. Per ottimizzare i programmi

Programmare non significa soltanto utilizzare correttamente i comandi offerti da 4th DIMENSION. Potete strutturare la vostra programmazione e pertanto guadagnare tempo, talvolta in misura considerevole.

Abbiamo già fornito un esempio di strutturazione dei programmi quando abbiamo creato la procedura *Mostra nazione* nel Capitolo 1 di questo Manuale.
Avrete senz'altro occasione di utilizzare spesso questa possibilità offerta dal linguaggio di 4th DIMENSION se creerete applicazioni complesse, o semplicemente applicazioni nelle quali dovrete ripetere le stesse serie di istruzioni.

1. Richiamare una procedura dall'interno di una procedura

Riprendiamo uno degli esempi.
Immaginate di aver scritto queste righe:

```
`Chiediamo il nome di un cliente
xNome:=Richiedi("Cliente da cercare")
 `Se la richiesta è confermata
Se(OK=1)
 `Cerchiamo il cliente richiesto
 CERCA([Clienti];[Clienti]Nome=xNome)
Fine se
```

E' molto probabile che avrete bisogno di queste righe in diversi punti, ad esempio per modificare o stampare una scheda cliente.
Per non dover ripetere queste righe ad ogni procedura, potrete allora creare una procedura che chiamerete, ad esempio, *CercaClienti,* e che verrà richiamata dalle altre procedure.
Questa procedura globale andrà ad aggiungersi alla vostra lista di comandi predeterminati di 4th DIMENSION.
Sarà scritta in corsivo e funzionerà come se si trattasse di un comando proprio di 4th DIMENSION.

In questo modo, quando scrivete la vostra procedura per modificare le schede clienti, o per stamparne la lista, potrete richiamare la procedura *CercaClienti* come se si trattasse di un normale comando nel programma, con la particolarità che si tratta di un comando su misura che compie un'azione ben specifica.
Pertanto, la sequenza completa effettuata da 4th DIMENSION è la seguente.
Immaginiamo una procedura ModifCliente che vada a cercare ad un certo punto il comando *CercaClienti.*

Procedura *ModifClienti*

La procedura *CercaClienti* è richiamata, eseguita ed al termine, ritorna automaticamente all'esecuzione del resto della procedura *ModifClienti.*

Allo stesso modo si potrà utilizzare questa procedura *CercaClienti* all'interno di una procedura StampaClienti.

Il vantaggio di una simile organizzazione è semplicissimo:
- La procedura si trova una volta soltanto nella base dati. Anziché riscrivere queste righe in ogni procedura e per diverse volte, è sufficiente richiamarle ogni qualvolta serviranno.
- Se vi fosse un errore all'interno della procedura, la correzione della procedura *CercaClienti* risolverebbe il problema in tutte le procedure in cui viene richiamata.

Se non aveste richiamato una procedura, sarebbe stato necessario correggere tutte le procedure contenenti queste poche righe.
- Queste procedure potranno servirvi come base di partenza per sviluppi futuri, riducendo i tempi di lavoro.

2. Passare un parametro ad una procedura

Oltre alla possibilità di richiamare una procedura, 4th DIMENSION vi permette di "*passare i parametri* " a questa procedura. Significa che potete trasferire dei valori in questa procedura, esattamente come fate quando richiamate un comando in 4th DIMENSION.

Immaginate di voler cercare le fatture insolute di un certo periodo per poi stamparle, visualizzarle in un'altra procedura, o esportarle in una terza procedura.
Questa ricerca comporta innanzitutto l'individuazione delle fatture insolute, e in seguito, tra le fatture trovate, la ricerca di quelle comprese tra le due date.
Questa fase di ricerca sarà identica per tutte le vostre procedure.
Potrete quindi estrarla e farne una procedura a parte, che chiameremo *CercaInsoluto*.
In tal caso, dobbiamo passare a questa nuova procedura le date d'inizio e di fine del periodo.
Scriveremo in ognuna delle procedure chiamanti *CercaInsoluto*(Inizio;Fine).

Quando l'esecuzione arriverà alla procedura *CercaInsoluto*, potrete accedere ai parametri ricevuti.
4th DIMENSION dispone di un potente strumento sintattico per definire un parametro il cui contenuto particolare è sconosciuto, ma che è sempre della stessa natura: il segno del dollaro, $, al quale si aggiunge un numero 1,2,3 ... n ...

Nel nostro caso, $1 sarà il valore d'inizio periodo e $2 sarà il valore di fine periodo.

Quindi, se $1 rappresenta l'inizio del periodo e $2 la fine, possiamo scrivere:

```
CERCA([Fatture];[Fatture]Saldo=0;*)
CERCA([Fatture];&[Fatture]Data>$1;*)
CERCA([Fatture];&[Fatture]Data<$2)
```

*Nota: il segno * significa che l'istruzione di ricerca viene estesa alle righe seguenti.*

Potete passare qualsiasi tipo di variabile e di campo ad una procedura.
Come vedremo in seguito, non è possibile passare allo stesso modo le variabili di tipo vettore.

3. Creare le proprie funzioni

Potete creare le vostre funzioni, esattamente come potete creare i comandi. Anche queste funzioni possono ricevere argomenti utilizzabili sotto forma di variabili $1...$n. Il valore restituito dalla funzione dovrà essere inserito nella variabile $0.

Esempio
Vogliamo creare una funzione che esegua l'azione seguente:
Vogliamo introdurre una data a partire dalla data del giorno proposta come valore standard. Vogliamo richiamare questa funzione scrivendo semplicemente:
LaData:=*RichiediData* ("Data di partenza")

La funzione *RichiediData* verrà scritta in questo modo:

```
 `$1 = testo della domanda
 `Visualizza la richiesta con la data del giorno come
 valore per difetto
xData:=Richiedi($1;Stringa(Data odierna))
 `se fatto clic su OK
Se(OK=1)
   `$0, il valore ritornato sarà la data inserita dall'uten-
   te
   $0:=Stringa in data(xData)
  `Se abbiamo fatto clic su Annulla
altrimenti
   `La nostra funzione restituisce una data nulla
  $0:=!00/00/00!
Fine se
```

Note:
- Una funzione non può restituire vettori. Se volete ottenere un vettore, utilizzate le variabili di tipo puntatore descritte più avanti in questo stesso manuale.
- Una funzione può dare un solo valore e se desiderate ottenere diversi valori, dovrete utilizzare variabili globali che verranno modificate durante la chiamata della vostra procedura.

4. Le variabili locali

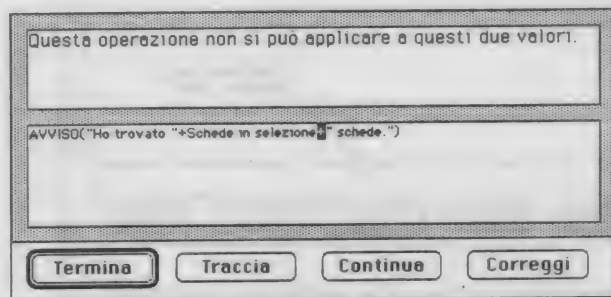
Gli argomenti ricevuti da una procedura hanno quindi il nome di $1,$2...$n.
Tali variabili vengono create soltanto durante l'esecuzione di questa procedura. Sono perciò chiamate *variabili locali* ed hanno una durata limitata nel tempo.

Potete definire voi stessi le variabili locali anteponendo al loro nome il carattere $. L'utilizzo di queste variabili locali si rivela interessante per i seguenti motivi:
- sono gestite, all'interno di 4th DIMENSION, in una lista separata con accesso diretto
- tali variabili esistono soltanto nella procedura attualmente eseguita, e vengono quindi eliminate automaticamente uscendo dalla procedura.

Note:
- Se definite delle variabili locali in una procedura di formato, tali variabili sono locali rispetto alla fase in cui sono definite, ossia la loro durata è di una fase (o in **Prima**, o in **Durante** o in **Dopo** , o **In intestazione**, o **In rottura**, o **In pié di pagina**). Non stupitevi quindi se una variabile locale inizializzata in Prima, risulta indefinita in Dopo.
- Potete definire variabili locali per i seguenti tipi: Numerico, Alfanumerico, Booleano, Data, Ora.
- Una variabile locale può essere un puntatore, ma un puntatore non può puntare verso una variabile locale.
- Non esistono variabili locali di tipo vettore.
- Se utilizzate $MiaVar in diverse procedure, consideratele come tante variabili diverse, in quanto $MiaVar è una variabile locale. In compenso, potete passare una variabile locale come argomento di una funzione.

8. Come usare il modo Traccia

1. Interruzione di una procedura in seguito ad un errore

Se viene incontrato un errore di programmazione, 4th DIMENSION presenta una finestra di questo tipo:

Nel rettangolo in alto si trova l'errore commesso e in quello inferiore l'istruzione che ha causato l'errore. Potete scegliere fra quattro possibilità.

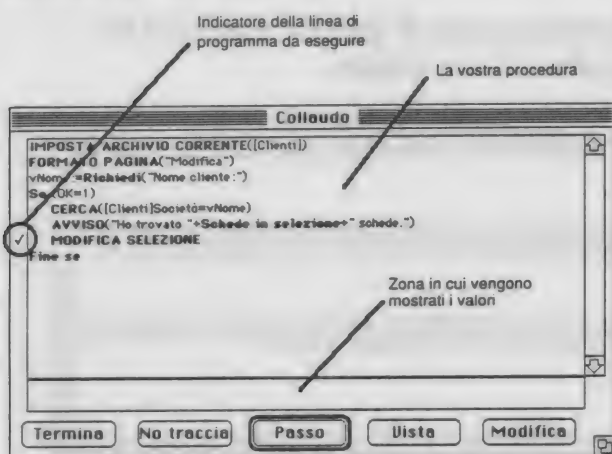
Termina: Questo pulsante arresta la procedura in atto e ritorna alla barra di menu principale o al formato corrente, se si tratta di una procedura di formato.

Traccia: Questo pulsante vi consente di passare al modo Traccia, ossia ad un ambiente che vi permette di comprendere più facilmente l'errore che è stato commesso.

Continua: Se fate clic su questo pulsante, 4th DIMENSION tenta di continuare ad eseguire il programma. Noterete spesso che l'errore avvenuto provoca altri errori nel proseguimento del programma.

Correggi: Se avete già capito di quale errore si tratta, potrete fare clic sul pulsante **Correggi**. 4th DIMENSION passa allora all'ambiente **Progetto** ed apre automaticamente la procedura che contiene l'errore.

2. Il modo Traccia

Entrando in modo Traccia, 4th DIMENSION presenta la seguente finestra:

Questa finestra presenta, nella parte superiore, la procedura che contiene l'errore. Noterete il carattere √ a destra della vostra procedura. Questo segno indica il punto in cui si trova l'esecuzione del programma.
Nella parte bassa, potrete, come vedremo, richiedere la visualizzazione di un valore qualsiasi.

Termina: Questo pulsante arresta la procedura in corso e ritorna alla barra di menu principale.

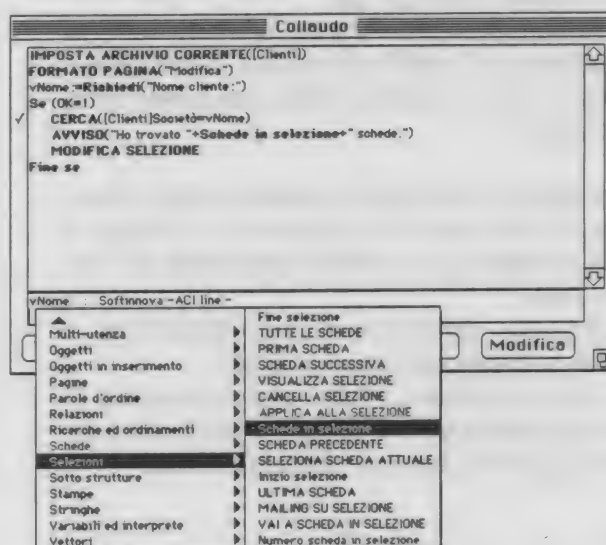
No traccia: Se fate clic su questo pulsante, 4th DIMENSION riprende regolarmente l'esecuzione del programma.
La finestra del modo Traccia resta sullo schermo fino a quando non verrà svolta un'azione che farà ridisegnare lo schermo.

Passo: Se fate clic su questo pulsante, 4th DIMENSION eseguirà la successiva riga di programma e si interromperà quando l'avrà completata.

Vista: Quando utilizzate il modo Traccia, la relativa finestra, ovviamente, nasconde quella di 4th DIMENSION.

Fate clic sul pulsante **Vista** per far passare in primo piano la finestra su cui stavate lavorando. Facendo nuovamente clic riapparirà la finestra del modo Traccia.

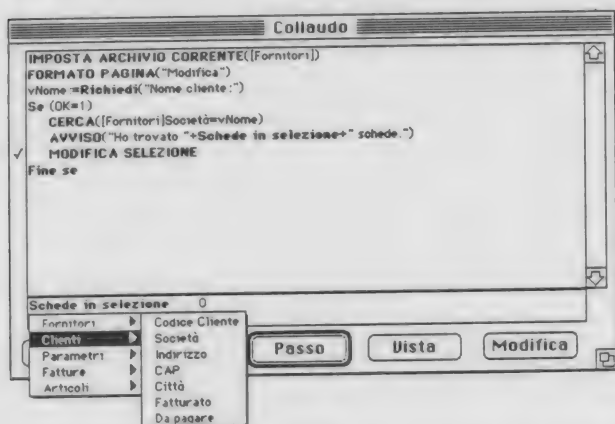
Modifica: Se avete capito l'errore, potete fare clic sul pulsante **Modifica**. 4th DIMENSION passerà quindi all'ambiente **Progetto**, aprendo automaticamente la procedura in cui si è verificato l'errore.

3. Area dei valori, visualizzazione di una variabile, di una funzione

Potete scrivere nella parte inferiore della finestra del modo Traccia il nome di una variabile o di una funzione di 4th DIMENSION.
Al momento dell'esecuzione del passo successivo, o facendo clic sulla procedura visualizzata, 4th DIMENSION mostrerà il contenuto della variabile o il risultato della funzione.

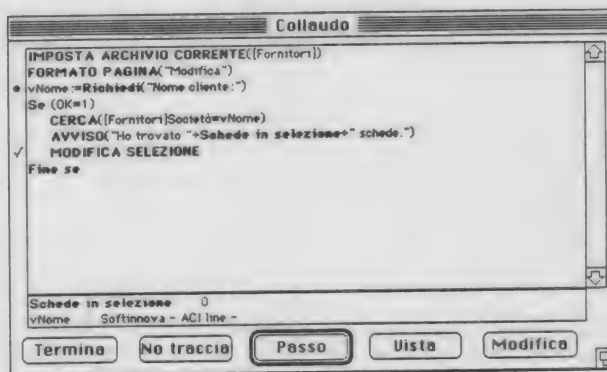
Per aumentare le dimensioni dell'area dei valori, fate clic sulla procedura visualizzata nella finestra del modo Traccia tenendo premuto il tasto **Opzione**.
La linea di separazione tra l'area che contiene il programma e l'area dei valori si posizionerà nel punto in cui avete fatto clic.

Per inserire un comando standard di 4th DIMENSION, potete anche richiamarne la lista. Per fare questo, dopo aver fatto clic nell'area dei valori ed ottenuto il cursore di inserimento, premete contemporaneamente il tasto **Comando** ed il bottone del mouse. Dopo qualche istante, apparirà la lista dei comandi standard ordinata per tipo. Per selezionare un comando specifico, selezionate un tipo sul menu a comparsa in modo da farne comparire il sottomenu collegato, quindi trascinate il puntatore a freccia sul comando desiderato. Quando rilascerete il bottone del mouse, il comando selezionato verrà inserito nella finestra del modo Traccia.

Nota
Alcuni comandi standard non possono essere eseguiti in modo Traccia. I comandi NUOVA FINESTRA o AGGIUNGI SCHEDA, ad esempio, hanno un funzionamento interno che non ne permette la chiamata.

4. Per visualizzare il valore di un campo

Per visualizzare il contenuto di un campo, inserite il nome del campo nell'area dei valori.
Il concetto di archivio in uso non viene usato esplicitamente nella finestra del modo Traccia: un campo dovrà essere sempre preceduto dal nome dell'archivio a cui appartiene. Il nome dell'archivio deve essere indicato tra parentesi quadre.

Nello stesso modo in cui visualizzate la lista dei comandi di 4th DIMENSION, potete anche visualizzare la lista degli archivi e dei campi che vi sono contenuti.
Ugualmente, dopo aver fatto clic nell'area dei valori ed ottenuto il cursore di inserimento, premete contemporaneamente il tasto **Opzione** ed il bottone del mouse.
Dopo qualche istante, comparirà la lista degli archivi, elencati in base al loro ordine di creazione.
Per effettuare la selezione di un campo specifico, selezionate l'archivio sul menu a comparsa, in modo da farne comparire il sottomenu collegato, quindi trascinate il puntatore a freccia sul campo desiderato. Quando rilascerete il bottone del mouse, il campo selezionato sarà visualizzato nella finestra del modo Traccia.

5. Introduzione di un punto di arresto

Per introdurre un punto d'arresto, vale a dire per ottenere l'esecuzione delle procedure da parte del programma fino al punto di interruzione, spostate il puntatore a freccia sul lato sinistro della finestra del modo Traccia. Il puntatore assumerà l'aspetto di un •.
Fate quindi clic alla sinistra di una riga della procedura per introdurre il punto di arresto.
Per fare eseguire la procedura, fate clic sul pulsante **No traccia.**
4th DIMENSION tornerà al modo Traccia non appena incontrerà la linea contrassegnata con •.

Questo metodo si rivela molto utile per seguire procedure contenenti iterazioni (**Ciclo**, **Ripeti**...). Per disattivare un punto di arresto, basta fare clic su di esso. E' possibile introdurre diversi punti d'arresto nella stessa procedura.

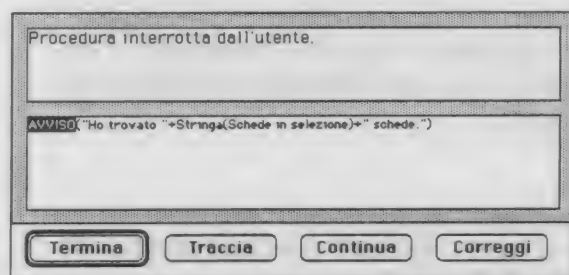
6. Step Over

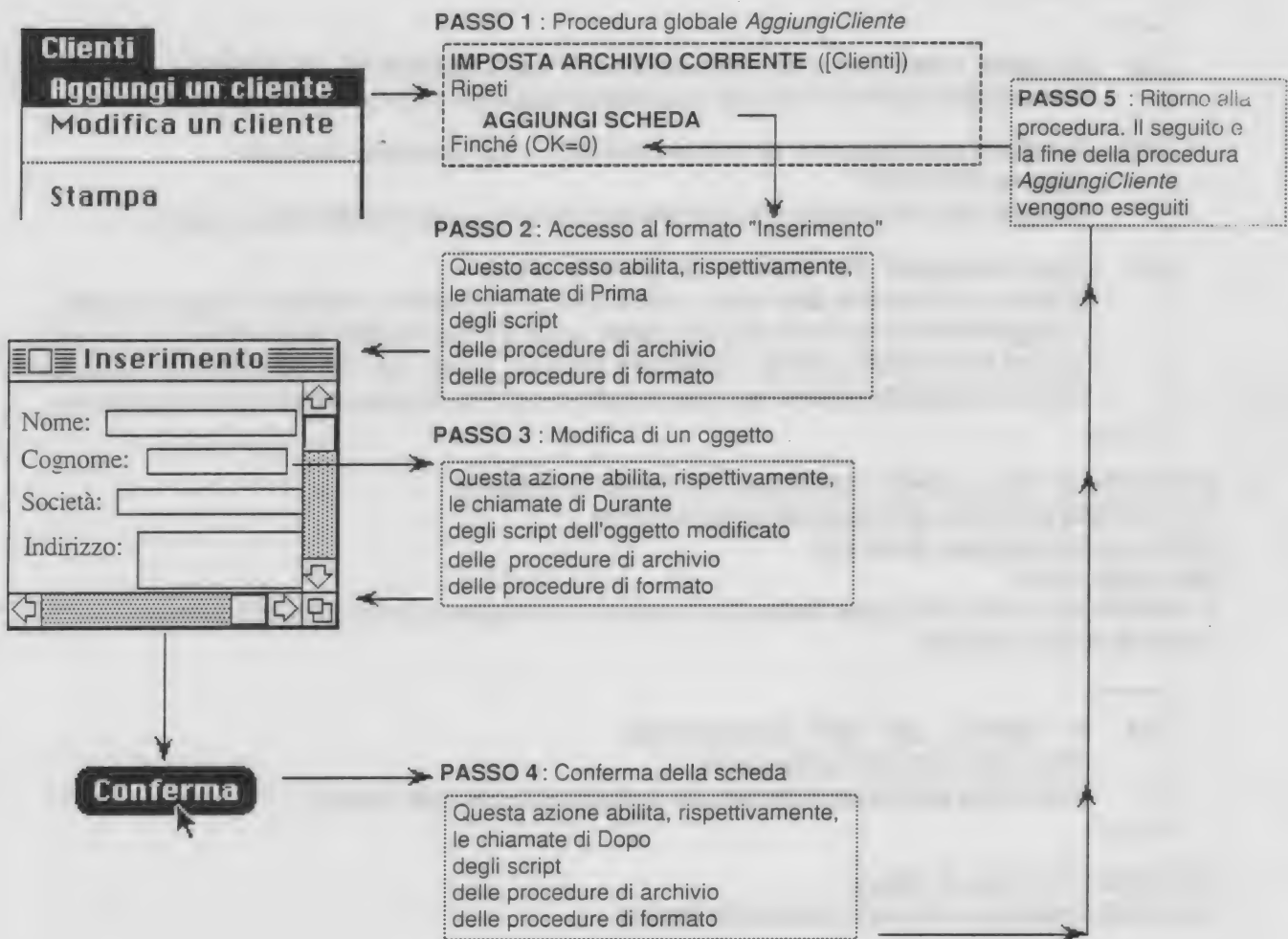
Durante l'esecuzione passo passo di una procedura (pulsante **Continua** o tasto **A-Capo**) può succedere che la procedura in corso ne richiami un'altra.
Se volete che 4th DIMENSION non esegua la procedura chiamata passo passo, ma consideri la chiamata di procedura come un passo unico, premete il tasto della **Maiuscole** durante l'esecuzione del passo in questione.

Se volete che questa funzione venga attivata nel corso dell'intera esecuzione della procedura, indipendentemente dal numero di procedure richiamate, premete il tasto **Blocco Maiuscole**.

7. Osservazioni generali

1. In modo Traccia, non potete seguire una procedura di interruzione su un evento. In effetti, gli eventi saranno completamente gestiti dal modo Traccia stesso.

2. Per interrompere una procedura in corso, premete il tasto Opzione e fate clic. Sullo schermo vedrete questa finestra:

Appendice
Architettura delle chiamate: i cicli di esecuzione

Immaginate di trovarvi nell'**ambiente Personalizzato**.
Avete creato una procedura di aggiunta clienti, che avete chiamato, ad esempio, *AggiungiClienti*.
Il percorso logico effettuato da 4th DIMENSION in funzione delle diverse chiamate sarà pertanto di questo tipo:

Se non avete personalizzato i vostri menu, siete nell'ambiente Utilizzo. Per l'esempio indicato qui sopra, avete soltanto i passi 2, 3 e 4.

1. Cicli di esecuzione in fase di inserimento o modifica (formato Pagine)

1. **Prima della comparsa del formato d'inserimento sullo schermo**
   Prima degli script relativi ad ogni pagina
   Prima della procedura di archivio
   Prima della procedura di formato del formato Pagina

2. **Modifica di una zona del formato**
   Durante dello script dell'oggetto modificato
   Durante della procedura di archivio
   Durante della procedura di formato del formato Pagina

   2a ***Se state inserendo una sotto-scheda o una scheda di un archivio incluso:***
   Prima della procedura di formato del formato Lista della sotto-struttura inclusa

   2b ***Se state modificando un sotto-archivio o un archivio incluso:***
   Durante dello script
   Durante della procedura di formato del formato Lista della sotto-struttura inclusa

   2c ***Aggiornamento del formato in questo caso:***
   Chiamata dei Durante degli script, delle procedure di archivio e della procedura di formato.
   L'aggiornamento avviene quando avete usato un accessorio della scrivania, se avete aperto e successivamente richiuso una finestra, e così via...4th DIMENSION richiama i cicli di esecuzione soltanto nel caso in cui una parte della zona debba essere aggiornata.

3. **Conferma di una scheda modificata al livello principale**
   Durante della procedura di formato del formato Pagina
   Durante della procedura di archivio
   Dopo degli script
   (Il Durante degli script non viene rieseguito, in quanto non esistono azioni su questi script al momento della conferma).

   3a ***In seguito, per ogni sotto-scheda:***
   Dopo degli script dei sotto-campi
   Dopo della procedura di formato del formato Lista del sotto-archivio

   Dopo della procedura di archivio
   Dopo della procedura di formato del formato Pagina

Note
Una scheda viene considerata modificata se avete modificato il valore di un campo dell'archivio a cui appartiene con la programmazione o inserendo un valore.
La sola modifica di una variabile di questo formato, o di un campo di un altro archivio, non è una condizione sufficiente per considerare la scheda in stato di modifica. Se avete necessità di questi due tipi di modifica, potete attuarli per mezzo della programmazione.
Inoltre, se confermate una scheda che non è in stato di modifica, questa non verrà riscritta.

2. Cicli di esecuzione in fase di visualizzazione (formato Lista)

1. **Prima chiamata di VISUALIZZA SELEZIONE**
   Prima degli script di tutte le pagine
   Prima della procedura di archivio
   Prima della procedura di formato del formato Pagina

2. **Visualizzazione della lista**
   In intestazione degli script e della procedura di formato, una volta.
   Prima e Durante (contemporaneamente) degli script e della procedura di formato per ogni scheda che compare sullo schermo.

3. **Doppio clic su una scheda**
   Durante del formato Lista
   Prima del formato Pagina (script e procedura di formato)

   3a ***Passaggio ad un'altra scheda***
   Durante del formato Pagina della scheda in uso (script e procedura di formato)
   Prima del formato Pagina della nuova scheda in uso (script e procedure di formato)

   3b ***Clic su Conferma nel formato Pagina***
   Durante del formato Pagina
   Chiamata delle fasi descritte per la visualizzazione della lista (2)

   Nota
   Se non siete in VISUALIZZA SELEZIONE, bensì in MODIFICA SELEZIONE, il percorso è lo stesso fino a questo punto, ma vi trovate nel caso della registrazione della scheda, descritto nella prima parte.

4. **Clic su un pulsante di Finito**
   Durante degli script
   Durante della procedura di formato

3. Cicli di esecuzione in fase di inserimento o modifica (formato Lista)

1. **Visualizzazione della lista**
   In intestazione degli script e della procedura di formato, una volta
   Prima e Durante (contemporaneamente) degli script e della procedura di formato per ogni scheda che compare sullo schermo.

2. **Modifica di una zona**
   Durante degli script e della procedura di formato

3. **Passaggio alla scheda successiva**
   Dopo degli script
   Dopo della procedura di archivio
   Dopo della procedura di formato.

4. Cicli di esecuzione durante la stampa

1. **Ad ogni inizio di pagina in stampa**
   In intestazione degli script e della procedura di formato

2. **Per ogni scheda**
   Prima degli script
   Prima della procedura di formato del formato Lista

   2a Per ogni sotto-scheda o scheda di un archivio incluso
   Prima degli script dei sotto-campi
   Prima della procedura di formato del sotto-campo

   Durante degli script
   Durante della procedura di formato del formato Lista

   2b Per ogni sotto-scheda o scheda di un archivio incluso
   Prima degli script dei sotto-campi
   Prima della procedura di formato del formato Lista della sotto-scheda
   Durante degli script dei sotto-campi
   Durante della procedura di formato del formato Lista della sotto-scheda

   Nota
   Il Prima può essere chiamato diverse volte. 4th DIMENSION gestisce queste chiamate in funzione dei calcoli di dimensionamento della pagina da effettuare, in particolare quando il formato comporta elementi di dimensioni variabili, come gli archivi inclusi, le sotto-strutture, le aree di testo o le immagini.

3. **Ad ogni rottura**
   In rottura del formato Lista

4. **Ad ogni cambio pagina**
   In pié di pagina del formato Lista

5. Cicli di esecuzione in importazione

1. **Ad ogni registrazione di schede**
   Dopo degli script
   Dopo delle procedure di archivio
   Dopo della procedura di formato del formato Pagina

6. Cicli di esecuzione in esportazione

1. **Ad ogni invio di scheda**
   Prima degli script
   Prima della procedura di formato del formato Lista

4th DIMENSION

4 . LINGUAGGIO

ACI

Suggerimenti per la lettura

Organizzazione di questa sezione

Questa sezione presenta l'analisi di ognuno dei comandi del linguaggio di 4th DIMENSION.
Tali comandi sono trattati per argomenti, in base alla classificazione fornita nel programma stesso.
Per il manuale tuttavia, non abbiamo adottato un ordine alfabetico, abbiamo invece scelto un ordine che riflette le priorità di utilizzazione. Ovviamente, quest'ordine è relativo e non ha molto significato per coloro che hanno familiarità con 4th DIMENSION.
La lettura di un manuale di riferimento è raramente lineare. Vi consigliamo tuttavia di esaminare velocemente tutti gli argomenti per avere una visione d'insieme del linguaggio e delle possibilità che offre.

Ogni argomento corrisponde a concetti fondamentali del programma.
Non abbiamo ripreso l'analisi dei concetti relativi ad ogni gruppo di comandi. Per questo, dovrete fare riferimento agli altri manuali.
E' ovvio, ad esempio, che l'uso dei comandi relativi alle selezioni presuppone la conoscenza del concetto di selezione in uso.

In alcuni casi, abbiamo introdotto un'analisi d'insieme dell'argomento trattato, mentre questa viene semplicemente accennata negli altri manuali.
Abbiamo introdotto queste considerazioni d'insieme relative a questi temi:
- la comunicazione *
- le direttive di compilazione
- la multi-utenza
- i vettori.

Nonostante non esistano comandi specifici per i puntatori, abbiamo introdotto l'argomento dei puntatori in questa sezione. In effetti, questo strumento estremamente potente di 4th DIMENSION presuppone una certa disinvoltura nell'uso del linguaggio nel suo insieme.

* Quando parliamo di comunicazioni in questi manuali, ci occupiamo soltanto della comunicazione tramite porta seriale. Non si tratta quindi di ciò che viene definito connettività.
Se desiderate informazioni sulle numerose possibilità di connettività che esistono in 4th DIMENSION, fate riferimento al vostro rivenditore di fiducia.

Simboli utilizzati in questo manuale

Questo manuale riprende tutti i comandi classificati per temi.
La composizione di ogni comando si presenterà nel modo seguente:

NOME DEI COMANDI

Sintassi

Sintassi del comando

Parametri

Elenco dei parametri	Tipo di parametro	Breve descrizione

Contesto

Eventuale contesto d'utilizzo

Descrizione

Descrizione del comando e del suo utilizzo

Esempio

Esempio di utilizzo del comando

Riferimenti

Eventuali comandi simili o collegati

Nella descrizione dei parametri, useremo le seguenti abbreviazioni:

Espr_Num Espressione di tipo numerico
Espr_NumPos Espressione di tipo numerico positivo
Espr_Alfa Espressione di tipo alfanumerico
Espr_Data Espressione di tipo data
Espr_Ora Espressione di tipo ora
Espr_Immagine Espressione di tipo immagine
Espr_Bool Espressione di tipo booleano
Espr_Vett Vettore o puntatore di vettore
Espr_Archivio Archivio o puntatore di archivio
Espr_Campo Campo o puntatore di campo
Espr_Var Variabile o puntatore di variabile
Espr_SottoArch Sotto-archivio o puntatore di sotto-archivio
Punt_Archivio Puntatore di archivio
Punt_Campo Puntatore di campo

➤ Valore restituito da una funzione

Tipologia dei comandi

* Non sono oggetto di comandi specifici, ma di un'analisi di uso.

Elenco dei comandi

Italiano	Inglese	N° pagina
Archivi		
IMPOSTA ARCHIVIO CORRENTE	DEFAULT FILE	54
FORMATO LISTA	OUTPUT LAYOUT	55
FORMATO PAGINA	INPUT LAYOUT	55
Schede in archivio	Records in file	56
Schede		
CARICA SCHEDA	LOAD RECORD	57
REGISTRA SCHEDA	SAVE RECORD	57
NUOVA SCHEDA	CREATE RECORD	58
CANCELLA SCHEDA	DELETE RECORD	59
DUPLICA SCHEDA	DUPLICATE RECORD	60
VAI A SCHEDA	GOTO RECORD	60
Numero scheda	Record number	61
Numero sequenziale	Sequence number	61
IMPILA SCHEDA	PUSH RECORD	62
DISIMPILA SCHEDA	POP RECORD	63
Inserimento		
AGGIUNGI SCHEDA	ADD RECORD	64
MODIFICA SCHEDA	MODIFY RECORD	65
DIALOGO	DIALOG	66
AGGIUNGI SOTTO_SCHEDA	ADD SUBRECORD	68
MODIFICA SOTTO_SCHEDA	MODIFY SUBRECORD	68
MODIFICA SELEZIONE	MODIFY SELECTION	69
Relazioni		
RELAZIONE A UNO	RELATE ONE	73
NUOVA SCHEDA SU RELAZIONE	CREATE RELATED ONE	74
REGISTRA RELAZIONE	SAVE RELATED ONE	74
RELAZIONE A UNO PRECEDENTE	OLD RELATED ONE	75
REGISTRA RELAZIONE PRECEDENTE	SAVE OLD RELATED ONE	76
ATTIVA RELAZIONE	ACTIVATE LINK	76
RELAZIONE A MOLTI	RELATE MANY	77
RELAZIONE A MOLTI PRECEDENTE	OLD RELATED MANY	78

Ricerche e ordinamenti

Selezioni

Sotto strutture

Cicli di esecuzione

Direttive di compilazione

Multi_utenza

IMPOSTA ARCHIVIO CORRENTE

DEFAULT FILE

Sintassi
IMPOSTA ARCHIVIO CORRENTE(Archivio)

Parametri	Tipo	Descrizione
Archivio	Espr_Archivio	Archivio da impostare come archivio corrente

Descrizione
Questo comando permette di semplificare la scrittura delle procedure. In effetti, molti comandi di 4th DIMENSION si riferiscono ad un archivio. L'uso del comando IMPOSTA ARCHIVIO CORRENTE evita di dover precisare ad ogni livello il nome dell'archivio.

Note
- IMPOSTA ARCHIVIO CORRENTE permette di omettere il parametro archivio passato ad una funzione o ad un comando, ma ciò non è valido per quanto riguarda i nomi dei campi. In altri termini, una chiamata IMPOSTA ARCHIVIO CORRENTE non vi dispensa dall'anteporre il nome dell'archivio ai nomi dei campi dell'archivio corrente.
- Non esiste alcun archivio corrente all'apertura della base dati
- Non può esistere più di un archivio corrente
- La chiamata di IMPOSTA ARCHIVIO CORRENTE non cambia l'archivio visualizzato all'apertura della base dati in uso.

Esempio
Questo esempio illustra la stessa procedura scritta a sinistra con il comando IMPOSTA ARCHIVIO CORRENTE e a destra scritta senza l'uso di questo comando. Potete notare che l'uso di IMPOSTA ARCHIVIO CORRENTE semplifica la scrittura delle vostre procedure.

```
IMPOSTA ARCHIVIO CORRENTE([Comandi])
TUTTE LE SCHEDE
CERCA
Se(OK=1)
  Se(Schede in selezione#0)
    FORMATO LISTA("LaLista")
    STAMPA SELEZIONE
  altrimenti
    AVVISO ("Nessuna scheda")
  Fine se
altrimenti
  AVVISO("Comando annullato")
Fine se
```

```
TUTTE LE SCHEDE([Comandi])
CERCA([Comandi])
Se(OK=1)
  Se(Schede in selezione([Comandi])#0)
  FORMATO LISTA([Comandi];"LaLista")
  STAMPA SELEZIONE([Comandi])
  altrimenti
  AVVISO("Nessuna scheda")
  Fine se
altrimenti
  AVVISO("Comando annullato")
Fine se
```

FORMATO LISTA **OUTPUT LAYOUT**

Sintassi
FORMATO LISTA({Archivio};Il Formato)

Parametri	Tipo	Descrizione
Archivio	Espr_Archivio	Archivio di cui si vuole cambiare il formato Lista
IlFormato	Espr_Alfa	Nome del Formato

Contesto
In qualsiasi contesto, eccetto durante la visualizzazione di *Archivio.*

Descrizione
FORMATO LISTA permette di impostare *IlFormato* quale formato Lista corrente per l'archivio *Archivio.* L'argomento *Archivio* è opzionale. Se viene omesso, FORMATO LISTA si riferirà all'archivio corrente.

Il formato Lista di un archivio viene utilizzato dai comandi VISUALIZZA SELEZIONE, MODIFICA SELEZIONE, STAMPA SELEZIONE, ESPORTA SYLK, ESPORTA DIF ed ESPORTA TESTO.

Esempi

FORMATO LISTA([Clienti];"Lista Clienti")

FORMATO LISTA([Clienti];"Lista"+**Stringa**(i))

IMPOSTA ARCHIVIO CORRENTE([Clienti])
FORMATO LISTA("Lista")

Riferimenti
IMPOSTA ARCHIVIO CORRENTE, FORMATO PAGINA

FORMATO PAGINA **INPUT LAYOUT**

Sintassi
FORMATO PAGINA({Archivio};Il Formato)

Parametri	Tipo	Descrizione
Archivio	Espr_Archivio	Archivio di cui si vuole modificare il formato pagina
Il Formato	Espr_Alfa	Nome del Formato

Contesto
In qualsiasi contesto, eccetto durante un inserimento in *Archivio.*

Descrizione
FORMATO LISTA permette di impostare *IlFormato* quale formato Lista corrente per l'archivio *Archivio*. L'argomento *Archivio* è opzionale. Se viene omesso, FORMATO LISTA si riferirà all'archivio corrente.

Il formato Lista di un archivio viene utilizzato dai comandi VISUALIZZA SELEZIONE, MODIFICA SELEZIONE, STAMPA SELEZIONE, ESPORTA SYLK, ESPORTA DIF ed ESPORTA TESTO.

Esempi

FORMATO LISTA([Clienti];"Lista Clienti")

FORMATO LISTA([Clienti];"Lista"+**Stringa**(i))

IMPOSTA ARCHIVIO CORRENTE([Clienti])
FORMATO LISTA("Lista")

Riferimenti
IMPOSTA ARCHIVIO CORRENTE, FORMATO PAGINA

Schede in archivio — Records in file

Sintassi
Schede in archivio({Archivio})
➤ Numerico

Parametri	Tipo	Descrizione
Archivio	Espr_Archivio	Archivio di cui volete conoscere il numero di schede

Descrizione
Schede in archivio è una funzione che vi restituisce il numero totale di schede attualmente registrate nell'archivio *Archivio*. *Archivio* è opzionale. Se questo argomento viene omesso, Schede in archivio si riferirà all'archivio corrente.

Nota
Schede in archivio è una funzione del tutto indipendente dalla selezione di *Archivio*. Se desiderate sapere il numero di schede selezionate, usate la funzione Schede in selezione.

Esempi

x:=**Schede in archivio**([Clienti])

IMPOSTA ARCHIVIO CORRENTE([Clienti])
x:=**Schede in archivio**

Riferimenti
Schede in selezione, Schede in insieme

CARICA SCHEDA LOAD RECORD

Sintassi
CARICA SCHEDA({Archivio})

Parametri	**Tipo**	**Descrizione**
Archivio	Espr_Archivio	Archivio della scheda da caricare

Contesto
Uso in Multi-utenza.

Descrizione
CARICA SCHEDA carica in memoria la scheda corrente dell'archivio *Archivio*. Questo comando è inutile in mono-utenza, in quanto 4th DIMENSION carica sempre automaticamente in memoria la scheda corrente. Questo comando è invece utile in multi-utenza. A questo proposito, consultate la sezione relativa alla programmazione in multi-utenza.
Archivio è opzionale. Se questo argomento viene omesso, CARICA SCHEDA si riferirà all'archivio corrente.

Esempio
Consultate la sezione Multi-utenza

Riferimenti
REGISTRA SCHEDA

REGISTRA SCHEDA SAVE RECORD

Sintassi
REGISTRA SCHEDA({Archivio})

Parametri	**Tipo**	**Descrizione**
Archivio	Espr_Archivio	Archivio della scheda da registrare

Descrizione
Il comando REGISTRA SCHEDA causa la registrazione su disco della scheda corrente dell'archivio *Archivio*. Archivio è opzionale. Se questo argomento viene omesso, REGISTRA SCHEDA si riferirà all'archivio corrente. Se non esiste una scheda corrente in *Archivio* (nessuna scheda selezionata), REGISTRA SCHEDA non esegue alcuna azione. La chiamata di REGISTRA SCHEDA permette, in particolare, di registrare una scheda modificata tramite una procedura, o il cui inscrimento è stato annullato.

Esempio

```
IMPOSTA ARCHIVIO CORRENTE ([Clienti])      Quale scheda vogliamo aggiornare?
x:=Richiedi("Nome del cliente?")           Se la domanda è confermata
Se(OK=1)
   CERCA([Clienti]Nome=x)                  Cerca la scheda
   Se(Schede in selezione=1)               Se trovi la scheda
      [Clienti]Fatturato:=0                Azzera Fatturato
      REGISTRA SCHEDA                      Registra la scheda
   Fine se
Fine se
```

Riferimenti
CARICA SCHEDA

NUOVA SCHEDA — CREATE RECORD

Sintassi
NUOVA SCHEDA({Archivio})

Parametri	Tipo	Descrizione
Archivio	Espr_Archivio	Archivio in cui creare la scheda

Descrizione
NUOVA SCHEDA crea in memoria una nuova scheda nell'archivio *Archivio* e la seleziona come scheda corrente. *Archivio* è opzionale. Se questo argomento viene omesso, NUOVA SCHEDA si riferirà all'archivio corrente.
Diversamente da AGGIUNGI SCHEDA, NUOVA SCHEDA non richiama la scheda in inserimento.
Questa scheda viene creata unicamente nella memoria centrale: dovete quindi chiamare esplicitamente la procedura REGISTRA SCHEDA se volete registrarla. Dopo una chiamata di NUOVA SCHEDA, la selezione corrente di *Archivio* rimane invariata, poichè la scheda è creata fuori dalla selezione. Quando invece registrerete la scheda per mezzo del comando REGISTRA SCHEDA, la selezione conterrà soltanto questa scheda.

Esempio
Procedura Startup: creazione di una scheda ad ogni ingresso nella base dati.

```
NUOVA SCHEDA([Parametri])
[Parametri]Data:=Data odierna
[Parametri]Utente:=Nome utente
[Parametri]Ora:=Ora attuale
REGISTRA SCHEDA([Parametri])
```

Riferimenti
REGISTRA SCHEDA

CANCELLA SCHEDA — DELETE RECORD

Sintassi
CANCELLA SCHEDA({Archivio})

Parametri	Tipo	Descrizione
Archivio	Espr_Archivio	Archivio in cui cancellare la scheda

Descrizione
CANCELLA SCHEDA elimina la scheda corrente dall'archivio *Archivio*. Se non esistono schede correnti, CANCELLA SCHEDA non esegue alcuna azione. *Archivio* è opzionale. Se questo argomento è omesso, CANCELLA SCHEDA si riferirà all'archivio corrente. La cancellazione della scheda è definitiva, non potete annullarla. Dopo una chiamata di CANCELLA SCHEDA, non esiste più selezione corrente.

Esempio

```
IMPOSTA ARCHIVIO CORRENTE([Clienti])
x:=Richiedi("Nome del cliente?")                      Quale scheda vogliamo cancellare?
Se(OK=1)                                              Se la domanda è confermata
   CERCA([Clienti]Nome=x)                             Cerca la scheda
   Se(Schede in selezione=1)                          Se trovi la scheda
   CONFERMA("Cancellazione scheda "+[Clienti]Nome)    Chiedi conferma della cancellazione
      Se(OK=1)                                        Se rispondo Sì
         CANCELLA SCHEDA                              Cancella la scheda.
      Fine se
   Fine se
Fine se
```

Ecco un esempio di procedura che vi permette di cancellare una scheda senza perdere la selezione corrente. A questo scopo utilizziamo due insiemi: "Selezione", contenente la scheda corrente e "DaCancellare", contenente la scheda da cancellare.

```
NUOVO INSIEME("Selezione")
NUOVO INSIEME VUOTO("DaCancellare")
AGGIUNGI AD INSIEME("DaCancellare")
DIFFERENZA("Selezione";"DaCancellare";"Selezione")
CANCELLA SCHEDA
USA INSIEME("Selezione")
ELIMINA INSIEME("Selezione")
ELIMINA INSIEME("DaCancellare")
```

Riferimenti
CANCELLA SELEZIONE

DUPLICA SCHEDA — DUPLICATE RECORD

Sintassi
DUPLICA SCHEDA({Archivio})

Parametri	Tipo	Descrizione
Archivio	Espr_Archivio	Archivio in cui duplicare la scheda

Descrizione
DUPLICA SCHEDA duplica la scheda corrente dell'archivio *Archivio*, ma non la chiama in inserimento, come AGGIUNGI SCHEDA. Potete chiamare DUPLICA SCHEDA in una procedura globale o nelle procedure di formato in Prima e Durante. Diversamente da NUOVA SCHEDA, questo comando non crea una scheda vuota, bensì una scheda contenente i valori di quella corrente. Come nel caso di NUOVA SCHEDA, dovete richiedere la registrazione della scheda con una chiamata di REGISTRA SCHEDA.

Esempio
Nella procedura di formato del formato d'inserimento clienti.

```
Se(Prima)
    CERCA([Clienti]Società="Dupont")
    DUPLICA SCHEDA
Fine se
```

Prima che appaia la scheda, cerchiamo la scheda di Dupont e la duplichiamo.

Riferimenti
NUOVA SCHEDA

VAI A SCHEDA — GOTO RECORD

Sintassi
VAI A SCHEDA({Archivio};NrAssoluto)

Parametri	Tipo	Descrizione
Archivio	Espr_Archivio	Archivio della scheda da destinare
NrAssoluto	Espr_NumPos	Numero della scheda

Descrizione
VAI A SCHEDA carica e seleziona la scheda dell'archivio *Archivio*, il cui numero assoluto è *NrAssoluto*. Questa scheda diventa la selezione corrente. Se *NrAssoluto* è inferiore al numero di scheda più piccolo, VAI A SCHEDA selezionerà la prima scheda di *Archivio*. Se *NrAssoluto* è superiore al numero di scheda più grande, VAI A SCHEDA selezionerà l'ultima scheda di *Archivio*.
Non confondete questo comando con quello di VAI A SCHEDA IN SELEZIONE, che si riferisce ad una posizione relativa all'interno della selezione corrente di *Archivio*.

Riferimenti
Numero scheda, Numero scheda in selezione, VAI A SCHEDA IN SELEZIONE

Numero scheda

Record number

Numero scheda({Archivio})
➤ Intero Lungo

Parametri	Tipo	Descrizione
Archivio	Espr_Archivio	Archivio della scheda di cui si vuole conoscere il numero

Descrizione
Numero scheda restituisce il numero interno della scheda corrente dell'archivio *Archivio*. Questo numero è indipendente dalla posizione della scheda all'interno della selezione corrente. Durante l'inserimento di schede, Numero schede restituisce il valore -3. Se non c'è nessuna scheda corrente, Numero scheda restituisce il valore -1.
Attenzione: questo numero non è fisso e può essere modificato con ORDINE PERMANENTE.

Riferimenti
Numero scheda in selezione, VAI A SCHEDA, VAI A SCHEDA IN SELEZIONE

Numero sequenziale

Sequence number

Sintassi
Numero sequenziale({Archivio})
➤ Intero Lungo

Parametri	Tipo	Descrizione
Archivio	Espr_Archivio	Archivio su cui effettuare la numerazione automatica

Descrizione
Numero sequenziale restituisce un Intero Lungo unico per ogni scheda aggiunta nell'archivio *Archivio*. Questa sequenza di numeri parte da 1 e viene incrementata di un'unità per ogni scheda inserita. Questa funzione è anche utilizzabile automaticamente durante la creazione di un formato, assumendo #N come valore per difetto in un campo di tipo Intero Lungo o Numerico.
In caso di utilizzazione in mono-utenza, potete chiamare questa funzione nel ciclo Prima. In caso di utilizzazione in multi-utenza, questa funzione dovrà essere chiamata nel ciclo Dopo, in quanto è possibile che un'altra scheda possa essere registrata mentre voi inserite la vostra.

Esempio

```
Se(Prima)
    ([Clienti]Codice_Cliente:=Numero sequenziale([Clienti])
Fine se
```

Prima che la scheda compaia sullo schermo, viene numerato il campo Codice_Cliente di [Clienti].

IMPILA SCHEDA — PUSH RECORD

Sintassi
IMPILA SCHEDA({Archivio})

Parametri	**Tipo**	**Descrizione**
Archivio	Espr_Archivio	Archivio della scheda da impilare

Descrizione
IMPILA SCHEDA impila in memoria una copia della scheda corrente di *Archivio*. Potrete in seguito recuperare questa scheda con il comando DISIMPILA SCHEDA. Ogni archivio comprende una pila di schede indipendenti, la cui capacità è limitata dalla memoria disponibile al momento.
Si tratta di una pila LIFO (Last In First Out).
I comandi IMPILA SCHEDA e DISIMPILA SCHEDA devono essere utilizzati in modo ragionevole, perchè un cattivo uso può determinare la saturazione della memoria.
Questi comandi si rivelano utili se desiderate esaminare l'archivio sul quale state effettuando l'inserimento. A questo scopo, impilate la scheda corrente, esaminate *Archivio* con i comandi di ricerca, di ordinamento, ..., poi disimpilate la scheda per ritornare a quella in corso di inserimento.
Attenzione: se impilate una scheda con lo scopo di cambiare la selezione corrente, non potrete più accedere ai campi della scheda impilata. Non dimenticate perciò di conservare nelle variabili quei valori di campo che potrebbero servirvi.
Le pile di schede di ogni archivio vengono cancellate ad ogni ritorno ai menu principali, al termine della vostra procedura.

Esempio
Realizziamo ora una procedura che ci avvertirà quando inseriremo il nome di una società già esistente nell'archivio. Si tratta soltanto di un avviso, e potremo decidere in seguito se l'inserimento è valido oppure no.
Procedura di formato del formato di inserimento

```
Se(Durante)
    Se(Modificato(Società))                         Se il campo Società è stato modificato
        vSocietà:=[Clienti]Società                  Registra il valore in una variabile temporanea
        IMPILA SCHEDA                               Impilia la scheda
        CERCA([Clienti]Società=vSocietà)            Controllia se esiste già una scheda con questo
        vRisult:=Schede in selezione                nome
        DISIMPILA SCHEDA                            Mantieni il risultato nella variabile vRisult
        Se(vRisult>0)                               Disimpila la scheda
            AVVISO("Già"+Stringa(vRisult)+"società")  Se vRisult fosse superiore a 0
        Fine se                                     Avviso: esistono già schede con questo nome
    Fine se
Fine se
```

Riferimenti
DISIMPILA SCHEDA

DISIMPILA SCHEDA

POP RECORD

Sintassi
DISIMPILA SCHEDA({Archivio})

Parametri	**Tipo**	**Descrizione**
Archivio	Espr_Archivio	Archivio della scheda da disimpilare

Descrizione
DISIMPILA SCHEDA disimpila l'ultima scheda dell'archivio *Archivio*, impilata con il comando IMPILA SCHEDA. Consultate il comando IMPILA SCHEDA per la descrizione del loro uso associato.

Esempio
Riferitevi all'esempio IMPILA SCHEDA

Riferimenti
IMPILA SCHEDA

AGGIUNGI SCHEDA

ADD RECORD

Sintassi
AGGIUNGI SCHEDA({Archivio};{*})

Parametri	Tipo	Descrizione
Archivio	Espr_Archivio	Archivio nel quale aggiungere la scheda
*	Costante	Assenza di barre di scorrimento in inserimento

Descrizione
AGGIUNGI SCHEDA crea in memoria una nuova scheda per l'archivio *Archivio*, la seleziona come scheda corrente dell'archivio e la propone per l'inserimento. AGGIUNGI SCHEDA utilizza il formato Pagina corrente dell'archivio.
Archivio è opzionale. Se omettete questo argomento, AGGIUNGI SCHEDA agirà sull'archivio corrente.
Durante l'inserimento, 4th DIMENSION esegue le procedure associate ad eventuali sotto-strutture ed alla scheda, come descritto nell'appendice relativa ai cicli d'inserimento, al termine della prima parte di questo Manuale.

Quando l'inserimento è ultimato:
- se il formato possiede i pulsanti predeterminati di 4th DIMENSION, potrete sapere quale pulsante è stato attivato con il test sulla variabile di sistema OK:
    - OK restitusce il valore 1 se avete fatto clic sul pulsante Conferma
    - OK restituisce il valore 0 se avete fatto clic sul pulsante Annulla;
- se il formato possiede pulsanti personalizzati, potrete sapere quale pulsante è stato attivato, verificando semplicemente se la variabile pulsante presenta il valore 1 oppure 0. Se nel vostro formato avete previsto un solo pulsante di conferma, potete anche verificare il valore di OK, che sarà uguale a 1.

Se confermate l'inserimento, la scheda sarà automaticamente registrata. Se non lo fate, la scheda non verrà registrata, ma resterà comunque in memoria finché non cambierete scheda corrente. Prima di questo cambio potete, se necessario, provocare la registrazione della scheda per mezzo del comando **REGISTRA SCHEDA**.
L'argomento opzionale * non visualizza le barre di scorrimento dalla finestra d'inserimento, evitando di far scorrere il formato se non è necessario.

Note
- Durante l'inserimento potete modificare la selezione corrente di *Archivio*, chiamando un comando del tipo SCHEDA SUCCESSIVA. In tal caso, la scheda in inserimento andrà perduta, a meno che non l'abbiate precedentemente registrata o impilata. Consultate l'ultima parte di questo argomento, relativa alle tecniche avanzate di inserimento.
- I pulsanti standard a sinistra sullo schermo non vengono mostrati quando il vostro formato contiene almeno una variabile di tipo Pulsante, Pulsante a Inversione, Pulsante Invisibile, Radio Bottone, Pulsante di Opzione o Radio Immagine.

Esempio
Chiamata di un inserimento di schede clienti

AGGIUNGI SCHEDA([Clienti])

Procedura di aggiunta schede clienti

IMPOSTA ARCHIVIO CORRENTE([Clienti])
Ripeti
  AGGIUNGI SCHEDA
Finché(OK=0)

Operazione ripetuta finché OK=0, ossia l'inserimento è annullato.

Riferimenti
MODIFICA SCHEDA, REGISTRA SCHEDA

MODIFICA SCHEDA — MODIFY RECORD

Sintassi
MODIFICA SCHEDA({Archivio};{*})

Parametri	Tipo	Descrizione
Archivio	Espr_Archivio	Archivio della scheda da modificare
*	Costante	Assenza di barre di scorrimento in inserimento

Descrizione
MODIFICA SCHEDA vi permette di modificare la scheda corrente dell'archivio *Archivio.*
MODIFICA SCHEDA utilizza il formato Pagina corrente dell'archivio. Se non c'è nessuna scheda corrente (non esistono schede selezionate), MODIFICA SCHEDA non compie alcuna azione.
Durante l'operazione d'inserimento, 4th DIMENSION esegue le procedure associate ad eventuali sotto-strutture ed alla scheda (consultate l'appendice relativa ai cicli d'inserimento, al termine della prima parte di questo Manuale).

Terminato l'inserimento, il valore restituito dal pulsante su cui è stato fatto clic è identico a quello di AGGIUNGI SCHEDA, descritto in precedenza. Se confermate l'inserimento, le modifiche vengono automaticamente registrate. Se non confermate l'inserimento, la scheda modificata non viene registrata, ma resterà comunque in memoria con le modifiche eseguite fino a quando non cambierete scheda corrente. Prima di questo cambiamento, potrete registrare le modifiche servendovi del comando **REGISTRA SCHEDA**.
Archivio è opzionale. Se omettete questo argomento, MODIFICA SCHEDA agirà sul'archivio corrente.
L'argomento opzionale * non visualizza le barre di scorrimento della finestra d'inserimento, evitando di far scorrere il formato se non è necessario.

Note
- Durante l'inserimento potete modificare la selezione corrente di *Archivio*, chiamando un comando del tipo SCHEDA SUCCESSIVA. In tal caso, la scheda in inserimento andrà perduta, a meno che non l'abbiate precedentemente registrata o impilata. Consultate l'ultima parte di questo argomento, relativa alle tecniche avanzate di inserimento.

- I pulsanti standard a sinistra sullo schermo non vengono mostrati quando il vostro formato contiene almeno una variabile di tipo Pulsante, Pulsante a Inversione, Pulsante Invisibile, Radio Bottone, Pulsante di Opzione o Radio Immagine.

Esempio

MODIFICA SCHEDA([Clienti])	Modifica della scheda corrente dell'archivio [Clienti]
IIMPOSTA ARCHIVIO CORRENTE([Clienti]) **CERCA**([Clienti]Nome="Dupont") **MODIFICA SCHEDA**	Ricerca e modifica della scheda del Sig. Dupont

Riferimenti
AGGIUNGI SCHEDA, REGISTRA SCHEDA

DIALOGO — DIALOG

Sintassi
DIALOGO({Archivio};Il Formato)

Parametri	**Tipo**	**Descrizione**
Archivio	Espr_Archivio	Archivio a cui appartiene IlFormato
IlFormato	Espr_Alfa	Nome del formato usato come finestra di dialogo

Descrizione
DIALOGO visualizza sullo schermo il formato *IlFormato* dell'archivio *Archivio*. Non si tratta di un inserimento di schede, ma di una finestra di dialogo, ossia un formato d'interfaccia.
Potete servirvi soltanto delle variabili inseribili. Naturalmente, potete includere nella finestra di dialogo pulsanti di ogni genere, aree di scorrimento, aree grafiche, aree esterne, e così via. I campi, se esistono, non sono inseribili, tuttavia vengono visualizzati con il loro valore corrente.

Archivio è opzionale. Se questo argomento viene omesso, DIALOGO chiama *IlFormato* dell'archivio corrente. Se non esiste nessuna variabile di tipo pulsante nel formato, 4th DIMENSION inserirà automaticamente un pulsante Conferma ed un pulsante Annulla in fondo alla finestra di dialogo.

- Conferma della finestra di dialogo: confermate la finestra di dialogo facendo clic su Conferma, premendo il tasto A-capo o facendo clic su un pulsante di conferma da voi definito. In tal caso, la variabile OK sarà impostata a 1.
- Annullamento della finestra di dialogo: annullate la finestra di dialogo facendo clic su Annulla, premendo Command-. o facendo clic su un pulsante di annullamento da voi definito. In questo caso, la variabile OK sarà impostata a 0.

Il test della variabile OK deve essere effettuato fuori dalla finestra di dialogo, nella procedura globale o nelle procedure di formato che danno origine alla chiamata della finestra di dialogo.

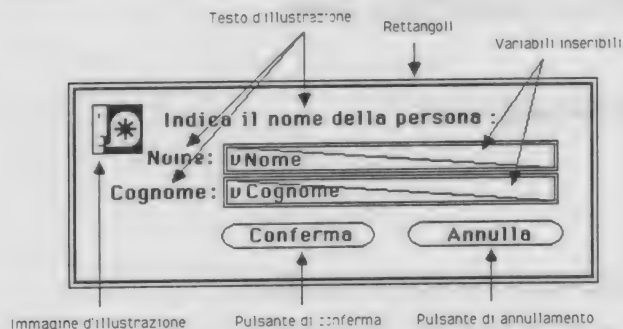
Esempio

In ambiente **Personalizzato**, con una scelta di menu vogliamo cercare una scheda nell'archivio [Personale], per poterla modificare. Scriviamo la procedura *ModifPers* associata alla scelta di menu:

```
IMPOSTA ARCHIVIO CORRENTE([Personale])
FORMATO PAGINA("Inserimento")
OK:=1
vCognome:=""
vNome:=""
Mentre(OK=1)
    DIALOGO("Ricerca")
    Se(OK=1)
        Se((vCognome#"")&(vNome#""))
            CERCA([Personale]Cognome=vCognome;*)
            CERCA(&[Personale]Nome=vNome)
            Se(Schede in selezione#0)
                MODIFICA SCHEDA
            altrimenti
                AVVISO("Nessuna persona trovata")
            Fine se
        Fine se
    Fine se
Fine mentre
```

Seleziona [Personale] come archivio corrente
Seleziona Inserimento come formato Pagina
Inizializza OK a 1 per il ciclo Mentre (OK=1)
Inizializza le variabili della finestra di dialogo come variabili alfanumeriche vuote

Visualizza la finestra "Ricerca" sullo schermo
Verifica OK all'uscita dalla finestra di dialogo
Se sono indicati dei valori in entrambe le variabili, cerca la scheda
Se trovi almeno una scheda autorizza la modifica.

Il formato "Ricerca" utilizzato come finestra di dialogo per la procedura *ModifPers*, è composto in questo modo:

Riferimenti

AGGIUNGI SCHEDA, MODIFICA SCHEDA

AGGIUNGI SOTTO_ SCHEDA — ADD SUBRECORD

Sintassi
AGGIUNGI SOTTO_SCHEDA(IlSottoarchivio;IlFormato;{*})

Parametri	**Tipo**	**Descrizione**
IlSottoarchivio	Espr_SottoArch	Sotto_Archivio in cui viene effettuato l'inserimento
IlFormato	Espr_Alfa	Formato da usare per l'inserimento
*	Costante	Assenza di barre di scorrimento in inserimento

Contesto
Inserimento in un formato

Descrizione
AGGIUNGI SOTTO_SCHEDA crea in memoria una nuova sotto-scheda di *IlSottoarchivio* per la scheda corrente dell'archivio al quale appartiene *IlSottoarchivio* e la propone per l'inserimento, utilizzando il formato Pagina *IlFormato*. Questa nuova sotto-scheda diventa la sotto-scheda corrente. Le procedure di formato, le procedure di archivio e gli script sono chiamati esattamente come in fase di inserimento.
Al termine dell'inserimento, potete sapere se è stato confermato, facendo un test sulla variabile OK che assumerà il valore 1. Dovrete poi chiamare il comando REGISTRA SCHEDA per registrare la scheda padre.
Nel caso delle sotto-strutture a più livelli, se non c'è una scheda corrente o una sotto-scheda corrente del livello superiore, AGGIUNGI SOTTO_SCHEDA non esegue alcuna azione.

Esempio

```
AGGIUNGI SOTTO_SCHEDA([Personale]Bambini;"FormatoBambini")
Se(OK=1)
    REGISTRA SCHEDA([Personale])
Fine se
```

Riferimenti
MODIFICA SOTTO_SCHEDA, NUOVA SOTTO_SCHEDA, REGISTRA SCHEDA

MODIFICA SOTTO_SCHEDA — MODIFY SUBRECORD

Sintassi
MODIFICA SOTTO_SCHEDA(IlSottoarchivio;IlFormato;{*})

Parametri	**Tipo**	**Descrizione**
IlSottoarchivio	Espr_SottoArch	Sotto-archivio nel quale avverrà l'inserimento
IlFormato	Espr_Alfa	Formato da usare per l'inserimento
*	Costante	Assenza di barre di scorrimento in inserimento

Contesto
Inserimento in un formato

Descrizione
MODIFICA SOTTO_SCHEDA propone la modifica della sotto-scheda corrente del sotto-archivio *IlSottoarchivio*. Il formato utilizzato sarà il formato *IlFormato*. Le procedure di formato, le procedure di archivio e gli script sono chiamati esattamente come in fase di inserimento.
Potete sapere se avete confermato l'inserimento verificando la variabile OK, che in tal caso avrà il valore 1. Se confermate l'inserimento, dovrete chiamare il comando REGISTRA SCHEDA per ottenere la registrazione della scheda padre insieme con le relative sotto-schede.
Se non esiste alcuna sotto-scheda corrente in *IlSottoarchivio*, MODIFICA SOTTO_SCHEDA non esegue alcuna azione.

Esempio
```
CERCA SOTTO_SCHEDE([Personale]Bambini;[Personale]Bambini'Nome="Paolo")
MODIFICA SOTTO_SCHEDA([Personale]Bambini;"FormatoBambini")
Se(OK=1)
    REGISTRA SCHEDA([Personale])
Fine se
```

Riferimenti
NUOVA SOTTO_SCHEDA, REGISTRA SCHEDA

MODIFICA SELEZIONE — MODIFY SELECTION

Sintassi
MODIFICA SELEZIONE({Archivio};{*})

Parametri	Tipo	Descrizione
Archivio	Espr_Archivio	Archivio contenente la selezione da modificare
*	Costante	Permette di ottenere la visualizzazione in formato Lista

Descrizione
MODIFICA SELEZIONE presenta sullo schermo le schede della selezione corrente dell'archivio *Archivio* in formato Lista. Contrariamente al comando VISUALIZZA SELEZIONE, se fate doppio clic su una scheda, questa verrà presentata per la modifica utilizzando il formato Pagina corrente. *Archivio* è opzionale. Se questo argomento viene omesso, MODIFICA SELEZIONE si riferirà all'archivio corrente.
Se la selezione corrente è vuota, un messaggio vi indica che nell'archivio non esistono schede selezionate.
L'argomento opzionale * si comporta esattamente come per VISUALIZZA SELEZIONE.

Esempio
Riferitevi all'esempio di VISUALIZZA SELEZIONE

Riferimenti
MODIFICA SCHEDA, VISUALIZZA SELEZIONE, USA INSIEME

Tecniche avanzate di inserimento

Questi esempi si riferiscono a concetti diversi dall'inserimento vero e proprio e si rivolgono unicamente agli utenti che hanno già familiarità con 4th DIMENSION.

Cambiamento della scheda corrente durante l'inserimento

Cambiamento della scheda corrente

Potete modificare la selezione corrente durante l'inserimento, ma dovrete assicurarvi di registrare le modifiche apportate alla scheda in inserimento. E' sufficiente creare un pulsante per il passaggio alla scheda successiva, il cui script sarà:

```
REGISTRA SCHEDA([MioArchivio])
SCHEDA SUCCESSIVA([MioArchivio])
```

Navigazione nella selezione (SlideShow)

Questa procedura associata ad un pulsante consente gli spostamenti da una scheda all'altra in fase di inserimento. Se quando arrivate alla scheda che vi interessa fate clic, 4th DIMENSION visualizzerà questa scheda sullo schermo.
Script del pulsante che attiva lo SlideShow:

```
IMPOSTA ARCHIVIO CORRENTE([Società])
CHIAMA SU EVENTO("StopEvent")               Installa la procedura d'interruzione.
StopEvt:=0                                  Inizializza la variabile di interruzione.
Se(Numero scheda nella selezione=Schede     Se sei sull'ultima scheda della selezione.
in selezione)
   PRIMA SCHEDA                             Ritorna alla prima scheda.
altrimenti
   SCHEDA SUCCESSIVA                        altrimenti passa alla scheda successiva.
Fine se
Mentre(StopEvt=0)                           Mentre il bottone del mouse è alzato.
   MOSTRA SCHEDA                            Visualizza la scheda corrente.
   Se(StopEvt=0)
      SCHEDA SUCCESSIVA                     Passa alla scheda successiva.
      Se(Fine selezione)                    Se giungi alla fine della selezione.
         PRIMA SCHEDA                       Ritorna alla prima scheda.
      Fine se
   Fine se
Fine mentre
CHIAMA SU EVENTO("")                        Disattiva la procedura d'interruzione.
```

Procedura StopEvent

```
StopEvent:=MouseDown
```

Ricerca in una selezione senza modificare la selezione

Procedura *Trova*

IMPOSTA ARCHIVIO CORRENTE([Società])	
x:=**Richiedi**("Società")	Richiedi quale società desidero cercare.
Se(OK=1)	
NUOVO INSIEME("InUso")	Registra la selezione corrente.
CERCA([Società];[Società]Società=x)	Cerca le società.
Se(Schede in selezione>0)	Se è stata trovata almeno una scheda.
NUOVO INSIEME("InSelezione")	Registra le schede trovate nell'insieme.
INTERSEZIONE("InSelezione";"InUso"; "InSelezione")	Mantieni soltanto le schede che appartengono anche a "InUso".
UNIONE("InSelezione";"InUso";"InUso")	Esegui l'unione di "InSelezione" e di "InUso".
USA INSIEME("InUso")	Utilizza l'insieme "InUso"
altrimenti	
USA INSIEME("InUso")	Non è stata trovata nessuna scheda, riprendi la selezione.
Fine se	
Fine se	

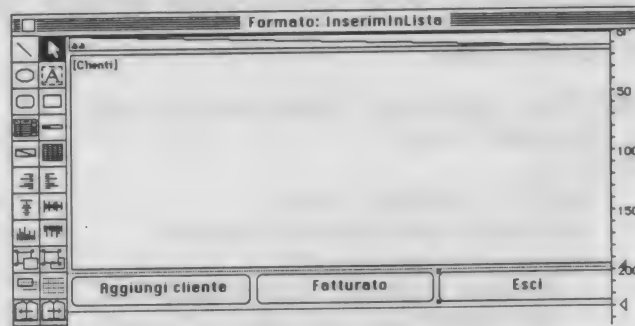
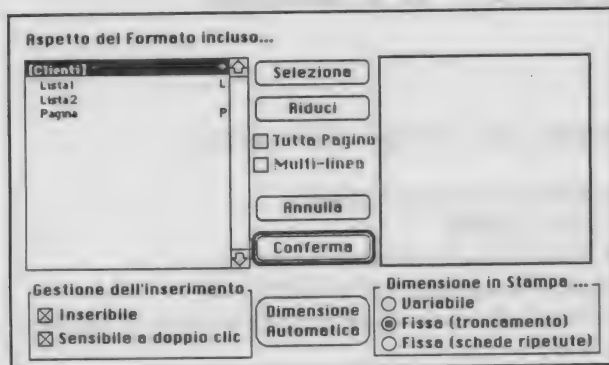
Inserimento di schede durante un'istruzione di VISUALIZZA o MODIFICA SELEZIONE

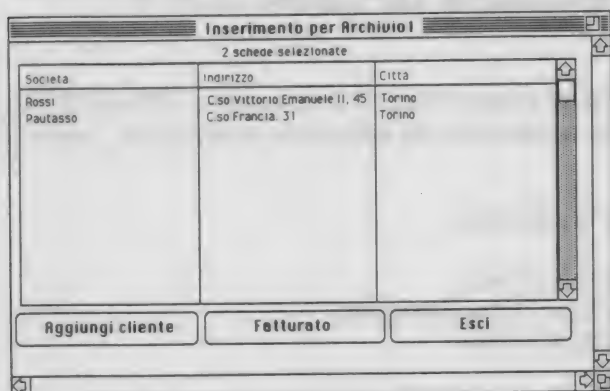
Potete chiamare il comando AGGIUNGI SCHEDA durante l'esecuzione di un'istruzione VISUALIZZA o MODIFICA SELEZIONE. In ogni caso, se desiderate conservare la selezione precedente, usate lo script che segue.

NUOVO INSIEME("InUso")	Registra la selezione
Ripeti	
AGGIUNGI SCHEDA	Aggiungi una scheda
Se(OK=1)	Se la scheda è stata confermata
AGGIUNGI AD INSIEME("InUso")	Aggiungila all'insieme
Fine se	
Finché(OK=0)	Ripeti l'operazione finché non viene annullato l'inserimento
USA INSIEME("InUso")	Usa l'insieme come selezione corrente.

Fare un inserimento in lista

Il modo inserimento in lista è accessibile soltanto nell'ambiente Utilizzo; comunque, potete in realtà riprodurre questo tipo d'inserimento. Se usate il comando MODIFICA SELEZIONE, le schede sono modificabili, ma soltanto in formato Pagina. Per ottenere l'inserimento in lista, dovrete creare un formato in un altro archivio contenente un archivio incluso. Questo si comporterà come un Sotto-archivio e sarà quindi inseribile o no, sensibile al doppio clic oppure no.
Create un formato contenente un formato di un archivio incluso, di grosse dimensioni. Scegliete i formati Lista e Pagina che desiderate.

L'opzione **Sensibile a doppio clic** permette di stabilire se autorizzate il passaggio in formato Pagina. I pulsanti nella parte inferiore del formato non vengono visualizzati nelle aree contenenti archivi inclusi. Metteteli nel formato padre.

Potete anche simulare un cambiamento di formato Lista creando un formato dotato di più pagine, contenente l'archivio incluso in ogni pagina, ma con un formato Lista diverso.

Create un pulsante a cui associerete un'azione di cambio pagina.
Nel nostro esempio, il pulsante **Fatturato** provoca il passaggio alla pagina successiva che comprende lo stesso archivio incluso, ma con un formato Lista contenente il fatturato e la data dell'ultima fattura dell'archivio clienti. Il risultato nell'inserimento sarà il seguente:

Il pulsante **Aggiungi cliente** vi permette di aggiungere un cliente in formato Pagina, grazie ad un breve script associato, che è il seguente:

```
NUOVO INSIEME([Clienti];"InUso")                Registra la selezione
Ripeti
   AGGIUNGI SCHEDA([Clienti])                   Aggiungi una scheda
   Se(OK=1)                                     Se la scheda è stata confermata
      AGGIUNGI AD INSIEME([Clienti];"InUso")    Aggiungila all'insieme
   Fine se
Finché(OK=0)                                    Ripeti l'operazione finchè l'inserimento non verrà annullato.
USA INSIEME([Clienti];"InUso")                  Usa l'insieme come selezione corrente.
```

Relazioni

RELAZIONE A UNO RELATE ONE

Sintassi
RELAZIONE A UNO({Archivio})
RELAZIONE A UNO(Chiamante;{Discriminante})

Parametri	**Tipo**	**Descrizione**
Chiamante	Espr_Campo	Campo da cui parte la relazione
Discriminante	Espr_Campo	Un campo dell'archivio di arrivo della relazione
Archivio	Espr_Archivio	Archivio da cui parte la relazione

Descrizione
RELAZIONE A UNO ha due diverse sintassi:

- Prima sintassi: RELAZIONE A UNO({Archivio})
Questa sintassi carica tutte le relazioni a uno automatiche di *Archivio.* Le relazioni vengono caricate seguendo lo schema di propagazione delle relazioni, come descritto nel Manuale *Ambiente Progetto.*
Non è necessario chiamare questo comando durante l'inserimento di una scheda di *Archivio*, in quanto le relazioni automatiche vengono già caricate da 4th DIMENSION. Se invece scrivete una procedura globale modificando una scheda di *Archivio* senza che questa compaia sullo schermo, le relazioni automatiche non vengono caricate. *Archivio* è un argomento opzionale. Se questo argomento viene omesso, RELAZIONE A UNO si riferirà all'archivio corrente.
Desideriamo stampare le schede dell'archivio Contatti, collegato all'archivio Imprese per mezzo del comando STAMPA FORMATO.

```
IMPOSTA ARCHIVIO CORRENTE([Contatti])
TUTTE LE SCHEDE                              Seleziona tutte le schede
ORDINA SELEZIONE([Contatti]Nome;>)           Ordina i contatti in ordine alfabetico
PARAMETRI DI STAMPA                          Chiama la finestra di dialogo per la stampa
Ripeti                                       Esamina le schede una ad una
    RELAZIONE A UNO                          Attiva le relazioni automatiche
    Caso fra
        :([Contatti]Tipo="Cliente")          Se si tratta di un cliente
        STAMPA FORMATO("Cliente")            Stampa con il formato "Cliente"
        :([Contatti]Tipo="Prospetto")        Se si tratta di un prospetto
        STAMPA FORMATO("Prospetto")          Stampa con il formato "Prospetto"
    Fine caso
    [Contatti]Stampato:=Vero                 Contrassegna la scheda come stampata
    REGISTRA SCHEDA                          Registra la scheda
    SCHEDA SUCCESSIVA                        Passa alla scheda successiva
Finché(Fine selezione)                       Ripeti l'operazione fino al termine della selezione
CAMBIO PAGINA                                Cambia pagina a fine stampa.
```

- Seconda sintassi: RELAZIONE A UNO(Chiamante;{Discriminante})
Questa sintassi provoca il caricamento della relazione che parte da *Chiamante*. RELAZIONE A UNO dà quindi luogo alla ricerca della scheda a cui punta la relazione ed al relativo caricamento in memoria. Questa scheda diventa così la scheda corrente dell'archivio chiamato. Se il vostro formato contiene dei campi dell'archivio chiamato, potrete aggiornarli chiamando RELAZIONE A UNO.
Se modificate alcuni valori della scheda chiamata, dovrete in seguito chiamare il comando REGISTRA RELAZIONE, descritto qui di seguito, per registrare le modifiche apportate.

Nel caso di una relazione non automatica, potete utilizzare il secondo argomento: *Discriminante*. Questo argomento deve essere un campo appartenente all'archivio che riceve la relazione. Se durante la chiamata della relazione 4th DIMENSION trova diverse schede possibili, verrà visualizzata una finestra contenente i valori del campo che riceve la relazione e del campo *Discriminante*. L'utente potrà quindi selezionare in questa lista la scheda sulla quale vorrà attivare la relazione, che diventerà la scheda corrente dell'archivio chiamato.

Riferimenti
REGISTRA RELAZIONE

NUOVA SCHEDA SU RELAZIONE — CREATE RELATED ONE

Sintassi
NUOVA SCHEDA SU RELAZIONE(Chiamante)

Parametri	Tipo	Descrizione
Chiamante	Espr_Campo	Campo da cui parte la relazione

Descrizione
NUOVA SCHEDA SU RELAZIONE agisce con le stesse modalità di RELAZIONE A UNO, ad eccezione del fatto che se la scheda puntata non esiste, questa verrà automaticamente creata in memoria. Dovrete in seguito compilarla tramite programmazione. Nessuno dei valori di questa scheda è compilato, perciò non dimenticate di assegnare il campo che riceve la relazione. Allo stesso modo, dovrete richiedere la registrazione di questa scheda chiamando il comando REGISTRA RELAZIONE.

Riferimenti
RELAZIONE A UNO, REGISTRA RELAZIONE

REGISTRA RELAZIONE — SAVE RELATED ONE

Sintassi
REGISTRA RELAZIONE(Chiamante)

Parametri	Tipo	Descrizione
Chiamante	Espr_Campo	Campo da cui parte la relazione

Descrizione
REGISTRA RELAZIONE permette di registrare una scheda precedentemente caricata con una chiamata del comando RELAZIONE A UNO o NUOVA SCHEDA SU RELAZIONE e provoca la registrazione della scheda puntata dalla relazione che parte da *Chiamante*, senza dover specificare a quale archivio appartiene. Quando gestite delle relazioni automatiche, il programma registra automaticamente le modifiche effettuate sulle schede collegate. Non occorre quindi chiamare il comando REGISTRA RELAZIONE. Se invece gestite un archivio collegato nell'ambito di una procedura globale, dovete chiamare il comando REGISTRA RELAZIONE per ottenere la registrazione delle schede collegate.
Se vogliamo aggiornare i nostri archivi di contatti, dovremo creare una procedura *Aggiornamento*.

```
IMPOSTA ARCHIVIO CORRENTE([Contatti])
TUTTE LE SCHEDE                                    Seleziona tutte le schede
Ripeti                                             Per tutte le schede
    RELAZIONE A UNO                                Carica le relazioni automatiche
    Se([Imprese]Tipo="Industria")                  Se Tipo è uguale a "Industria"
       Se([Contatti]Funzione="Dir.Produzione")     Se la funzione è "DirProduzione"
          [Impresa]NrDir:=[Impresa]NrDir+1         Incrementa il campo NrDir
          REGISTRA RELAZIONE (Codice_ Società)     Registra la scheda collegata
       Fine se
    Fine se
    SCHEDA SUCCESSIVA                              Passa alla scheda successiva
Finché(Fine selezione)                             Ripeti l'operazione per tutte le schede contatti.
```

Nota
Questa funzione deve essere usata soltanto all'interno di una relazione che punta al campo di un archivio.Se la vostra relazione punta ad un sotto-archivio, sarà sufficiente registrare la scheda che contiene la vostra sotto-scheda.

Riferimenti
RELAZIONE A UNO, NUOVA SCHEDA SU RELAZIONE

RELAZIONE A UNO PRECEDENTE — OLD RELATED ONE

Sintassi
RELAZIONE A UNO PRECEDENTE(Chiamante)

Parametri	**Tipo**	**Descrizione**
Chiamante	Espr_Campo	Campo da cui parte la relazione

Descrizione
Questa procedura è identica al comando RELAZIONE A UNO, ad eccezione del fatto che viene caricata la scheda a cui puntava la relazione precedente, vale a dire la relazione relativa all'ultima versione registrata della scheda Chiamante.

Esempio e riferimenti
Fate riferimento all'esempio di REGISTRA RELAZIONE PRECEDENTE

REGISTRA RELAZIONE PRECEDENTE — SAVE OLD RELATED ONE

Sintassi
REGISTRA RELAZIONE PRECEDENTE(Chiamante)

Parametri	Tipo	Descrizione
Chiamante	Espr_Campo	Campo da cui parte la relazione

Descrizione
Questa procedura è identica al comando REGISTRA RELAZIONE e permette di registrare una scheda precedentemente caricata con il comando RELAZIONE A UNO PRECEDENTE.

Esempio
Immaginate una struttura costituita da 2 archivi: Fatture e Clienti. Confermando la fattura, dovrete aggiornare il fatturato del vostro cliente.
La fattura 1001 ha un importo di 100.000 £; dovete quindi aggiungere 100.000 £ al campo Fatturato dell'archivio Clienti. Immaginate ora di modificare la fattura 1001 e che il suo importo, dopo la modifica, sia di 150.000 £. Aggiungete quindi 150.000 £ al campo Fatturato relativo al vostro cliente, che ammonterà a 250.000 £. Questo importo è errato, il valore esatto è di 150.000 £. Durante l'aggiornamento del fatturato, avreste dovuto detrarre innanzi tutto il valore precedente della fattura, poi aggiungere quello nuovo.
Allo stesso modo, se modificando la fattura aveste cambiato il cliente destinatario, avreste ottenuto un fatturato errato per il primo cliente. Ecco perché dovete avere la possibilità di riprendere la scheda collegata precedente se effettuate operazioni cumulative su un archivio collegato.

```
Se(Dopo)
    RELAZIONE A UNO PRECEDENTE(CodiceCliente)
    [Clienti]Fatturato:=[Clienti]Fatturato-Precedente(TotaleLordo)
    REGISTRA RELAZIONE PRECEDENTE(CodiceCliente)
    RELAZIONE A UNO(CodiceCliente)
    [Clienti]Fatturato:=[Clienti]Fatturato+TotaleLordo
    REGISTRA RELAZIONE(CodiceCliente)
Fine se
```

Richiama la scheda collegata precedente
Detrai il precedente valore del TotaleLordo
Registra nuovamente la scheda cliente precedente
Chiama la nuova scheda cliente
Aggiorna il fatturato
Registra la scheda

Riferimenti
RELAZIONE A UNO PRECEDENTE

ATTIVA RELAZIONE — ACTIVATE LINK

Nota
Questo comando esiste ancora nel linguaggio di 4th DIMENSION unicamente per motivi di compatibilità con le versioni precedenti. Nella nuova struttura di relazioni di 4th DIMENSION non è di alcuna utilità.

RELAZIONE A MOLTI — RELATE MANY

Sintassi
RELAZIONE A MOLTI({Archivio})
RELAZIONE A MOLTI(Chiamato)

Parametri	Tipo	Descrizione
Chiamato	Espr_Campo	Campo che riceve una relazione
Archivio	Espr_Archivio	Archivio che riceve le relazioni

Descrizione
RELAZIONE A MOLTI presenta due diverse sintassi:

- RELAZIONE A MOLTI({Archivio})
Questa sintassi provoca il caricamento delle relazioni a molti che puntano a *Archivio*; modifica perciò le selezioni correnti di tutti gli archivi che abbiamo relazioni a molti automatiche su *Archivio*. *Archivio* è un argomento opzionale. Se questo argomento viene omesso, RELAZIONE A MOLTI si riferirà all'archivio corrente.

- RELAZIONE A MOLTI(Chiamato)
Questa sintassi provoca il caricamento della relazione a molti che punta a *Chiamato*. Questa relazione può essere automatica o manuale.

La chiamata del comando RELAZIONE A MOLTI è necessaria se volete provocare esplicitamente il caricamento delle relazioni a molti. Può verificarsi in due casi:
- Aggiungete una scheda tramite programmazione e volete riattivare la relazione a molti
- Gestite delle schede tramite programmazione.

Esempio
Immaginate una struttura contenente un archivio Contatti collegato con l'archivio Imprese. Il formato d'inserimento dell'archivio Imprese contiene un formato incluso, in cui compaiono tutti i contatti di questa impresa. Durante l'inserimento, desideriamo aggiungere delle schede di contatti aprendo una finestra. Se non riattiviamo le relazioni a molti dopo la chiusura della finestra, l'archivio incluso conterrà soltanto l'ultimo contatto aggiunto.

Script del pulsante AggiungiCont nel formato inserimento dell'archivio Imprese.

NUOVA FINESTRA(50;50;450;300;0;"Inserimento Contatto")	Apri una finestra per l'inserimento
AGGIUNGI SCHEDA([Contatti])	Aggiungi una scheda contatto
CHIUDI FINESTRA	Chiudi la finestra
RELAZIONE A MOLTI([Imprese])	Attiva le relazioni a molti per aggiornare l'archivio incluso nel formato Pagina di Imprese.

Riferimenti
RELAZIONE A MOLTI PRECEDENTE

RELAZIONE A MOLTI PRECEDENTE — OLD RELATED MANY

Sintassi
RELAZIONE A MOLTI PRECEDENTE(Chiamato)
RELAZIONE A MOLTI PRECEDENTE({Archivio})

Parametri	Tipo	Descrizione
Chiamato	Espr_Campo	Campo che riceve una relazione
Archivio	Espr_Archivio	Archivio che riceve le relazioni

Descrizione
RELAZIONE A MOLTI PRECEDENTE agisce con le stesse modalità di RELAZIONE A MOLTI, tranne che attiva le relazioni a molti della versione registrata della vostra scheda.

Esempio
Se autorizzate la modifica di un campo che riceve una relazione, è importante conservare le schede collegate. A questo proposito, potete chiamare il comando RELAZIONE A MOLTI PRECEDENTE, che vi darà la selezione delle schede collegate in base al valore precedente. Potrete allora aggiornare i campi chiamanti.
Script del campo Codice

```
Se(Durante)
    Se(Codice#Valore precedente(Codice))                              Se Codice è stato modificato
       RELAZIONE A MOLTI PRECEDENTE(Codice)                           Carica le schede precedenti
       APPLICA ALLA SELEZIONE([Contatti];[Contatti]CodiceE:=Codice)   Modifica il campo CodiceE
    Fine se
Fine se
```

Riferimenti
RELAZIONE A MOLTI

Ricerche e ordinamenti

CERCA | SEARCH

Sintassi

CERCA({Archivio};Formula;{*})
CERCA({Archivio};Operatore_Campo;Comparatore;Valore;{*})

Parametri	Tipo	Descrizione
Archivio	Espr_Archivio	Archivio in cui viene creata la selezione
Operatore_Campo	Espr_Campo	Operatore e campo sul quale viene effettuata la ricerca
Comparatore	Espr_Alfa	<,<=,>,>=,=,# comparatori utilizzati
Valore	Espr	Valore da ricercare
*		Simbolo di ricerca da completare con un'istruzione successiva

Descrizione

CERCA permette di riprodurre con un'istruzione tutte le funzionalità della finestra di ricerca dell'ambiente Utilizzo. Costruite una sequenza di ricerca e ne richiedete l'esecuzione.
In questo modo, la ricerca nell'archivio [Persone] di persone che si chiamano "Rossi" o "Bianchi" e che abitano a "Torino" verrà scritta come segue:

```
IMPOSTA ARCHIVIO CORRENTE([Persone])
CERCA([Persone]Nome="Rossi";*)            Ricerca Nome="Rossi"
CERCA(| [Persone]Nome="Bianchi";*)        Oppure Ricerca Nome="Bianchi"
CERCA(& [Persone]Città="Torino")          E anche Ricerca Città="Torino"
```

- Prima sintassi: CERCA({Archivio};Formula;{*})
Archivio determina l'archivio nel quale verrà creata la selezione.
Formula è composta da un operatore logico (&;#;|), dal nome di un campo preceduto dal nome dell'archivio, da un comparatore (=;#;<;<=;>;>=) e da un valore cercato.
L'argomento * permette di estendere la formula di ricerca alle righe successive.

```
IMPOSTA ARCHIVIO CORRENTE([Persone])
CERCA([Persone]Salario>=10000;*)          Salari compresi tra 10000 e 50000
CERCA(&[Persone]Salario<50000)
```

Ogni *Formula*, eccetto la prima, comincia con un operatore logico:
& per E anche
| per Oppure (Opzione-:)
per Eccetto (Opzione-Maiuscole-§)
Si tratta della relazione logica tra ogni riga di ricerca.

L'operatore è seguito dal campo sul quale volete effettuare la ricerca. Questo campo non appartiene necessariamente all'archivio *Archivio*, nel quale viene effettuata la selezione.
Potete, infatti, eseguire una ricerca utilizzando le relazioni automatiche.

Immaginate la struttura seguente:

Cerchiamo le persone il cui nome inizia con "D" e che appartengono ad una società di Torino:

CERCA([Persone];[Persone]Nome="D@";*)	Cerca i nomi che iniziano con "D"
CERCA([Persone];&[Società]Città="Torino")	E anche Cerca le persone che lavorano in una società nella città="Torino"

Questa ricerca crea una selezione anche in base ad un altro archivio. Attenzione: quando eseguite una ricerca su relazione, la selezione corrente dell'archivio collegato viene modificata dalla vostra ricerca.

Se desiderate scrivere diverse righe di ricerca, la vostra sequenza dovrà terminare con il segno *. La ricerca verrà eseguita quando 4th DIMENSION incontrerà una riga di ricerca che non contiene *. Ad ogni archivio è associata una pila di righe di ricerca in corso, che viene mantenuta fino a quando non richiederete l'esecuzione della ricerca o uscirete da 4th DIMENSION.
La chiamata del comando CERCA senza argomenti o con il nome dell'archivio come unico argomento, dà luogo alla visualizzazione della finestra di dialogo standard di ricerca, se prima non avete passato nessuna riga , oppure effettua la ricerca se esiste almeno una riga di ricerca nella pila.

```
IMPOSTA ARCHIVIO CORRENTE([Persone])
vcer:=Richiedi("Cv Persone")
Se(OK=1)
    CERCA([Persone]CV=vcer;*)
Fine se
Mentre(OK=1)
    vcer:=Richiedi("Cv Persone")
    Se(OK=1)
        CERCA(|[Persone]CV=vcer;*)
    Fine se
Fine Mentre
CERCA
```

- Seconda sintassi: **CERCA**({Archivio};Operatore_Campo; Comparatore;Valore;{*})
La seconda sintassi di CERCA riprende gli stessi elementi della prima, ma permette di isolare ogni elemento della riga di ricerca. Questa sintassi vi consente, ad esempio, di creare sequenze di ricerca parametrizzabili, facendo variare il comparatore. Usando questa sintassi, potete passare *Valore* come espressione alfanumerica, e 4th DIMENSION effettuerà automaticamente la conversione del tipo in base al tipo di campo della ricerca.

Note
- Il comando CERCA non permette di effettuare ricerche calcolate. Per questo scopo, servitevi del comando CERCA CON FORMULA.
- L'utilità di questo comando rispetto a CERCA CON FORMULA sta nel fatto che 4th DIMENSION ottimizza le ricerche. Inoltre, gli operatori logici E anche, Oppure ed Eccetto, vengono gestiti automaticamente da 4th DIMENSION.
- Al momento dell'esecuzione della ricerca, 4th DIMENSION visualizza un messaggio che vi fornisce lo stato di avanzamento della ricerca. Potete cancellare questa visualizzazione con il comando DISATTIVA MESSAGGI, descritto al capitolo **Messaggi**.
- *Archivio* è un argomento opzionale. Se lo omettete, la selezione verrà creata all'interno dell'archivio corrente.
- E' obbligatorio anteporre ai campi il nome del loro archivio.
- Se *Valore* è un'espressione, questa viene valutata soltanto una volta prima dell'esecuzione della ricerca. Se desiderate effettuare una ricerca su un'espressione calcolata per ogni scheda, usate il comando CERCA CON FORMULA.

Note per la versione 1.0 x / versione 4.x.x
Il comando CERCA della versione 1.0.x di 4th DIMENSION si chiama ora CERCA CON FORMULA ed è descritta qui di seguito.

Riferimenti
TUTTE LE SCHEDE, CERCA CON FORMULA, CERCA CON INDICE, CERCA IN SELEZIONE

CERCA CON FORMULA — SEARCH BY FORMULA

Sintassi
CERCA CON FORMULA({Archivio};{Formula})

Parametri	Tipo	Descrizione
Archivio	Espr_Archivio	Archivio sul quale viene effettuata la ricerca
Formula	Espr_Bool	Espressione vera o falsa

Descrizione
CERCA CON FORMULA è un comando di ricerca che crea una selezione nell'archivio *Archivio*; questa selezione conterrà tutte le schede che soddisfano la condizione espressa in *Formula*. Una volta effettuata la ricerca, la prima scheda della nuova selezione diventa la scheda corrente di *Archivio* e viene caricata in memoria.
- *Archivio* è opzionale. Se questo argomento viene omesso, CERCA CON FORMULA si riferirà all'archivio corrente.
- Anche *Formula* è opzionale. Se questo argomento viene omesso, verrà presentata sullo schermo la finestra di ricerca standard dell'ambiente Utilizzo. Se la ricerca è effettuata per mezzo della finestra standard, potete sapere se avete annullato o interrotto la ricerca verificando la variabile OK, che in tal caso sarà uguale a 0; se la ricerca è terminata, la variabile OK sarà uguale a 1.

CERCA CON FORMULA dà luogo ad una ricerca sequenziale all'interno dell'archivio. Tutte le schede dell'archivio vengono verificate una dopo l'altra; il tempo di ricerca è quindi proporzionale al numero di schede presenti nell'archivio.

Esempi

CERCA CON FORMULA([Base Docum];[Base Docum]Parola Chiave="Margherita")
La selezione archivio [Base Docum] diventa l'insieme delle schede contenenti "Margherita" nel campo Parola Chiave.

CERCA CON FORMULA((**Sotto_stringa**([Cliente]CAP;1;2)="10")|(**Sotto_stringa**([Cliente]CAP;1;2)="20"))
Se l'archivio [Cliente] è l'archivio corrente, la selezione diventa l'insieme dei clienti situati nella zona di Torino e di Milano.

CERCA CON FORMULA([Leggi];[Leggi]Testo=LaVar)
- se LaVar è uguale a "Codice stradale@", la selezione diventa l'insieme delle leggi il cui testo inizia con "Codice stradale".
- se LaVar è uguale a "@furto@", la selezione diventa l'insieme delle leggi il cui testo contiene almeno una volta la parola "furto".
- se LaVar è uguale a "@regolamento", la selezione diventa l'insieme delle leggi il cui testo finisce con la parola "regolamento".

CERCA CON FORMULA([Fatture];LaFormula)
con la funzione *LaFormula* come

```
    RELAZIONE A UNO([Fatture]RifCliente)
    $0:=(Sotto_stringa([Cliente]CAP;1;2)="10")
```

La selezione dell'archivio [Fatture] diventa l'insieme delle fatture emesse ai clienti torinesi.

Riferimenti
TUTTE LE SCHEDE, CERCA, CERCA IN SELEZIONE, CERCA CON INDICE

CERCA IN SELEZIONE SEARCH SELECTION

Sintassi
CERCA IN SELEZIONE({Archivio};Formula)

Parametri	Tipo	Descrizione
Archivio	Espr_Archivio	Archivio sul quale viene effettuata la ricerca
Formula	Espr_Bool	Espressione che restituisce un valore vero o falso

Descrizione
CERCA IN SELEZIONE esegue le stesse operazioni di CERCA CON FORMULA, con la differenza che la ricerca viene effettuata soltanto nella selezione corrente di *Archivio* e non in tutto l'archivio. In questo modo vengono ottimizzati i tempi di ricerca, soprattutto nel caso di ricerche sequenziali. Durante la ricerca, 4th DIMENSION visualizza una finestra contenente un indicatore di progressione. Se nell'archivio non esistono schede selezionate, CERCA IN SELEZIONE non esegue alcuna azione.

Esempio
Supponiamo di avere un archivio di fatture con un campo [Fatture]Data indicizzato ed un campo [Fatture]Saldo non indicizzato. Vogliamo trovare l'insieme delle fatture insolute di un periodo in particolare (usiamo vInizio e vFine per indicare le due date che delimitano il periodo):

- Prima soluzione:

```
CERCACON FORMULA([Fatture];([Fatture]Data>=vInizio)&([Fatture]Data<=vFine)
                     &([Fatture]Saldo#0))
```

In questo modo otterrete certamente le fatture, ma scorrendo l'archivio in modo sequenziale, senza servirvi dell'indice sul campo [Fatture]Data.

- Seconda soluzione:

```
CERCA([Fatture]Data>=vInizio;*)
CERCA([Fatture]Data<=vFine)
CERCA IN SELEZIONE([Fatture]Saldo#0)
```

La ricerca viene attuata dapprima per indice su tutte le fatture del periodo e, successivamente, in modo sequenziale, ma soltanto nella selezione delle fatture che appartengono al periodo cercato: il vantaggio sarà tanto più rilevante quanto minore è il numero delle fatture corrispondenti a tale periodo.

Riferimenti
TUTTE LE SCHEDE, CERCA CON FORMULA, CERCA, CERCA CON INDICE

CERCA CON INDICE — SEARCH BY INDEX

Sintassi
CERCA CON INDICE({Campo1=Espr}|{Campo2±Limite_Inf;Limite_Sup};...)

Parametri	**Tipo**	**Descrizione**
Campo1	Espr_Campo	Campo preceduto dal nome del proprio archivio
Espr	Espr	Valore da ricercare
e/o		
Campo2	Espr_Campo	Campo preceduto dal nome del proprio archivio
Limite_Inf, Limite_Sup	Espr	Limiti dei valori da ricercare

Note per la versione 1.0.x/versione 4.x.x
Questo comando esiste ancora nel linguaggio di 4th DIMENSION per motivi di compatibilità con le versioni precedenti. E' sostituito per maggior convenienza dai comandi CERCA e CERCA CON FORMULA. Vi consigliamo di non utilizzarlo per assicurare la compatibilità delle vostre procedure con le future versioni di 4th DIMENSION.

Descrizione

CERCA CON INDICE modifica la selezione dell'archivio a cui appartengono i campi citati. Dopo una chiamata di CERCA CON INDICE, la selezione dell'archivio diventa l'insieme delle schede che soddisfano i test.
La prima scheda della nuova selezione diventa la scheda corrente dell'archivio e viene caricata in memoria.
I campi citati devono essere indicizzati ed il test può essere soltanto un test di uguaglianza, con un'espressione dello stesso tipo (l'opzione ± permette una ricerca all'interno di questi valori, il carattere ± è ottenuto scrivendo Opzione-Maiuscole-+).
Se tutti gli argomenti sono omessi, per l'archivio corrente viene presentata sullo schermo la finestra standard di ricerca con indice, accessibile in ambiente Utilizzo.
Se la ricerca viene effettuata per mezzo della finestra standard di ricerca con indice, potete sapere se avete annullato la ricerca con un test sulla variabile OK, che in tal caso sarà uguale a 0. Se la ricerca è stata eseguita, la variabile OK sarà uguale a 1.
Se sono indicati diversi test, 4th DIMENSION effettua tra di loro un E anche logico.
CERCA CON INDICE dà luogo ad una ricerca indicizzata all'interno dell'archivio. I tempi di ricerca sono molto brevi e praticamente indipendenti dal numero delle schede: circa 45 centesimi di secondo se viene trovata una sola scheda. Se vengono trovate più schede, questo tempo aumenta in modo logaritmico.

Esempi

CERCA CON INDICE([Ordini]RifCliente="Rossi")
La selezione dell'archivio [Ordini] diventa l'insieme degli ordini emessi dal cliente "Rossi".

CERCA CON INDICE([Fatture]RifCliente="Rossi";[Fatture]Pagata="Non pagata")
La selezione dell'archivio [Fatture] diventa l'insieme delle fatture non pagate dal cliente "Rossi".

CERCA CON INDICE([Fatture]Data±!10/10/1985!;Data odierna)
La selezione dell'archivio [Fatture] diventa l'insieme delle fatture emesse tra il 10/10/1985 (compreso) e la data odierna (compresa).

CERCA CON INDICE([Persone]Zona="Nord"[Persone]Salario±1000000;3000000)
La selezione dell'archivio [Persone] diventa l'insieme delle persone che lavorano al Nord ed il cui salario va da 1000000 (compreso) a 3000000 (compresi).

Riferimenti

TUTTE LE SCHEDE, CERCA, CERCA CON FORMULA, CERCA IN SELEZIONE, CERCA SU ESEMPIO

CERCA SU ESEMPIO — SEARCH BY LAYOUT

Sintassi
CERCA SU ESEMPIO({Archivio};{*})

Parametri	Tipo	Descrizione
Archivio	Espr_Archivio	Archivio di ricerca
*	Costante	Assenza di barre di scorrimento in inserimento

Descrizione
Il comando CERCA SU ESEMPIO riproduce le stesse funzioni della finestra di ricerca disponibile nell'ambiente Utilizzo. Il formato utilizzato sarà il formato pagina corrente. Se passate la costante * come argomento, non verranno visualizzate le barre di scorrimento. *Archivio* è opzionale. Se questo argomento viene omesso, CERCA SU ESEMPIO si riferirà all'archivio corrente.

Esempio

CERCA SU ESEMPIO([Clienti])

Riferimenti
TUTTE LE SCHEDE, CERCA, CERCA CON FORMULA, CERCA IN SELEZIONE, CERCA CON INDICE

ORDINA SELEZIONE — SORT SELECTION

Sintassi
ORDINA({Campo1;Ordine1};...;{Campo10;Ordine10})
ORDINA({Archivio})

Parametri	Tipo	Descrizione
Campo(n)	Espr_Campo	Campo dell'ennesimo ordinamento
Ordine(n)	>o<	Tipo di ordinamento dell'ennesimo ordinamento

Descrizione
Il comando ORDINA ordina la selezione dell'archivio del *Campo1* nella sequenza stabilita da *Ordine1*. Potete specificare un ordinamento a più livelli, utilizzando diversi campi di ordinamento all'interno dello stesso comando. Se volete effettuare un ordinamento crescente, utilizzate il simbolo ">". Se volete effettuare un ordinamento decrescente, utilizzate il simbolo "<". Se non precisate il senso, l'ordinamento sarà crescente.
Al termine dell'ordinamento, 4th DIMENSION carica la prima nuova scheda della selezione, impostandola come scheda corrente.
Dovete sempre anteporre a *Campo* il nome dell'archivio a cui appartiene, anche nelle procedure di formato.

Se utilizzate la seconda sintassi, 4th DIMENSION vi proporrà la finestra di dialogo standard per l'ordinamento.
Con entrambe le sintassi, 4th DIMENSION visualizzerà una finestra contenente l'indicatore di progressione dell'operazione. Potete evitare l'apertura di questa finestra con il comando DISATTIVA MESSAGGI prima di chiamare ORDINA.
La variabile OK sarà uguale a 1 al termine dell'ordinamento se questo si è svolto correttamente e se non avete fatto clic sul pulsante Interrompi. Negli altri casi, OK assumerà il valore di 0.
Se ordinate in base ad un solo criterio indicizzato, 4th DIMENSION effettua un ordinamento indicizzato più rapido rispetto ad uno sequenziale. Se effettuate un ordinamento su un campo indicizzato e se meno del 10% del totale delle schede appartiene alla selezione corrente, ORDINA effettuerà un ordinamento sequenziale, in questo caso più rapido.
Se ordinate in base a diversi criteri, l'ordinamento sarà sempre sequenziale.

Esempio

ORDINA([Indirizzi]CodicePostale;>;[Indirizzi]Nome;>)

ORDINA([Indirizzi])

ORDINA([Fatture];[Clienti]Nome;>)

Riferimenti
ORDINA CON FORMULA, ORDINA CON INDICE

ORDINA CON INDICE — SORT BY INDEX

Sintassi
ORDINA CON INDICE(Campo;{Ordine})

Parametri	Tipo	Descrizione
Campo	Espr_Campo	Campo indicizzato di ordinamento
Ordine	>o<	Sequenza di ordinamento

Nota per la versione 1.0.x / versione 4.x.x
Questo comando esiste ancora nel linguaggio di 4th DIMENSION per motivi di compatibilità con le versioni precedenti. Per maggior convenienza, è sostituito dai comandi ORDINA e ORDINA CON FORMULA. Vi consigliamo di non utilizzarlo per assicurarvi la compatibilità delle vostre procedure con le future versioni di 4th DIMENSION.

Descrizione
ORDINA CON INDICE ordina la selezione corrente dell'archivio di *Campo* nella sequenza stabilita da *Ordine*, utilizzando l'indice di *Campo*. ORDINA CON INDICE è in grado di effettuare ordinamenti su un solo campo per volta. Se *Ordine* è ">", l'ordinamento sarà crescente, altrimenti se *Ordine* è "<" l'ordinamento sarà decrescente. Se non precisate il senso, l'ordinamento sarà crescente. Al termine dell'ordinamento, 4th DIMENSION carica la prima scheda nuova della selezione, impostandola come scheda corrente. Se non avete interrotto lo svolgimento dell'operazione, la variabile OK assumerà il valore di 1, altrimenti OK assumerà il valore di 0.

Per il comando ORDINA CON INDICE, gli argomenti sono obbligatori.
Durante l'ordinamento, 4th DIMENSION visualizza una finestra contenente un indicatore di progressione; potete evitare l'apertura di questa finestra con il comando DISATTIVA MESSAGGI prima di chiamare ORDINA CON INDICE.

Esempio

ORDINA CON INDICE([Clienti]Società;>)

Riferimenti
ORDINA SELEZIONE, ORDINA CON FORMULA

ORDINA CON FORMULA — SORT BY FORMULA

Sintassi
ORDINA CON FORMULA(Archivio;Espressione1;{Ordine1};...;{Espressione10};{Ordine10})

Parametri	**Tipo**	**Descrizione**
Archivio	Espr_Archivio	Archivio a cui appartiene la selezione da ordinare
Espressione1...10	Espr	Valore di ordinamento
Ordine1...10	> o <	Senso di ordinamento

Descrizione
ORDINA CON FORMULA ordina la selezione corrente di *Archivio* in funzione dei valori calcolati delle espressioni.
Se *Ordine* è ">", l'ordinamento sarà crescente, altrimenti se *Ordine* è "<" l'ordinamento sarà decrescente. Se non precisate il senso, l'ordinamento sarà crescente.
Al termine dell'ordinamento, 4th DIMENSION carica la prima scheda nuova della selezione, impostandola come scheda corrente. Se non avete interrotto lo svolgimento dell'operazione, la variabile OK assumerà il valore di 1, altrimenti OK assumerà il valore di 0.
Per il comando ORDINA CON FORMULA, gli argomenti sono obbligatori.
Durante l'ordinamento, 4th DIMENSION visualizza una finestra contenente un indicatore di progressione; potete evitare l'apertura di questa finestra con il comando DISATTIVA MESSAGGI prima di chiamare ORDINA CON FORMULA.

Esempio

ORDINA CON FORMULA([Parole Chiave];**Lunghezza**([Parole Chiave]Parole);[Parole Chiave]Parole)

ORDINA CON FORMULA([Documenti];**Sotto_schede in selezione**([Documenti]Parole))

ORDINA CON FORMULA([Documenti];*NOccorrenze*("Karl Marx")

Immaginiamo la seguente procedura *NOccorrenze:*

```
$Testo:=[Documenti]descrizione
Ripeti
    $Pos:=Posizione($1;$Testo)
    Se($Pos#0)
        $0:=$0+1
        $Testo:=Sotto_stringa($Testo;Posizione($1;$Testo)+Lunghezza($1))
    Fine se
Finché($Pos=0)
```

Riferimenti
ORDINA SEQUENZA, ORDINA CON INDICE, ORDINE PERMANENTE

ORDINE PERMANENTE — SORT FILE

Sintassi
ORDINE PERMANENTE(Archivio;Espressione1;{Ordine1};...;{Espressione10};{Ordine10})

Parametri	Tipo	Descrizione
Archivio	Espr_Archivio	Archivio a cui appartiene la selezione da ordinare
Espressione1...10	Espr	Valore di ordinamento
Ordine1...10	>o<	Senso di ordinamento

Contesto
Soltanto in Mono-utenza

Descrizione
Il comando ORDINE PERMANENTE è esattamente identico al comando ORDINA CON FORMULA. L'ordinamento effettuato con questo comando sarà invece permanente, vale a dire il vostro archivio conserverà quest'ordine. L'ordinamento permanente dà luogo ad una nuova indicizzazione dell'archivio ordinato.

Esempi

ORDINE PERMANENTE([Clienti];[Clienti]Nome;>)

ORDINE PERMANENTE([Fatture];[Clienti]Nome;>)

ORDINE PERMANENTE([Parole Chiave];**Lunghezza**([Parole Chiave]Parole);[Parole Chiave]Parole)

Riferimenti
ORDINA SELEZIONE, ORDINA CON INDICE

TUTTE LE SCHEDE — ALL RECORDS

Sintassi
TUTTE LE SCHEDE({Archivio})

Parametri	Tipo	Descrizione
Archivio	Espr_Archivio	Archivio del quale volete selezionare tutte le schede

Descrizione
TUTTE LE SCHEDE seleziona la totalità delle schede dell'archivio *Archivio*. La prima scheda dell'archivio diventa la scheda corrente e viene caricata in memoria. *Archivio* è opzionale. Se questo argomento viene omesso, TUTTE LE SCHEDE avrà effetto sull'archivio corrente.

Nota
Non è necessario chiamare TUTTE LE SCHEDE prima della chiamata di un comando di ordinamento, come, ad esmpio, CERCA. I comandi di ricerca di 4th DIMENSION si applicano sempre alla totalità dell'archivio ad eccezione, ovviamente, dell'istruzione CERCA IN SELEZIONE.

Esempio

TUTTE LE SCHEDE([Clienti])

IMPOSTA ARCHIVIO CORRENTE([Clienti])
TUTTE LE SCHEDE

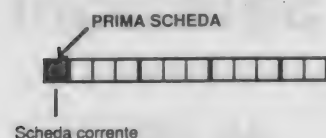
Riferimenti
CERCA CON FORMULA, CERCA, CERCA CON INDICE, CERCA IN SELEZIONE

PRIMA SCHEDA — FIRST RECORD

Sintassi
PRIMA SCHEDA({Archivio})

Parametri	Tipo	Descrizione
Archivio	Espr_Archivio	Archivio contenente la selezione

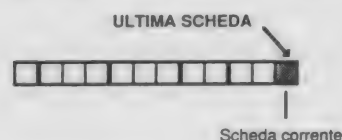
Descrizione
PRIMA SCHEDA imposta la prima scheda della selezione di *Archivio* come scheda corrente e la carica in memoria. *Archivio* è opzionale. Se questo argomento viene omesso, PRIMA SCHEDA avrà effetto sull'archivio corrente.

Se non esistono schede nella selezione oppure se la scheda corrente è la prima scheda della selezione, PRIMA SCHEDA non eseguirà alcuna azione.

Riferimenti
ULTIMA SCHEDA, Inizio selezione

ULTIMA SCHEDA — LAST RECORD

Sintassi
ULTIMA SCHEDA({Archivio})

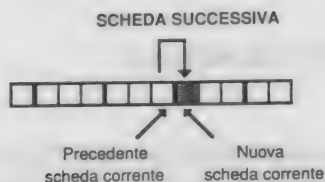
Parametri	Tipo	Descrizione
Archivio	Espr_Archivio	Archivio contenente la selezione

Descrizione
ULTIMA SCHEDA imposta l'ultima scheda della selezione di *Archivio* come scheda corrente e la carica in memoria. *Archivio* è opzionale. Se questo argomento viene omesso, ULTIMA SCHEDA avrà effetto sull'archivio corrente.
Se non esistono schede nella selezione oppure se la scheda corrente è l'ultima scheda della selezione, ULTIMA SCHEDA non eseguirà alcuna azione.

Riferimenti
PRIMA SCHEDA, Fine selezione

SCHEDA SUCCESSIVA — NEXT RECORD

Sintassi
SCHEDA SUCCESSIVA({Archivio})

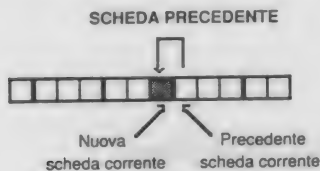
Parametri	Tipo	Descrizione
Archivio	Espr_Archivio	Archivio contenente la selezione

Descrizione
SCHEDA SUCCESSIVA imposta la scheda successiva a quella corrente nell'archivio *Archivio* come scheda corrente,e la carica in memoria. Se nella selezione non esiste alcuna scheda successiva, la funzione Fine selezione restituirà il valore Vero (funzione descritta in questo capitolo) e non avrete più alcuna scheda corrente, poichè vi trovate oltre la fine della selezione.
Se non esiste nessuna scheda corrente o nessuna selezione, SCHEDA SUCCESSIVA non eseguirà alcuna azione. *Archivio* è opzionale. Se questo argomento viene omesso, SCHEDA SUCCESSIVA si riferirà all'archivio corrente.

Riferimenti
SCHEDA PRECEDENTE, Fine selezione

SCHEDA PRECEDENTE — PREVIOUS RECORD

Sintassi
SCHEDA PRECEDENTE({Archivio})

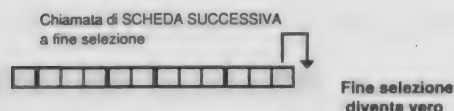
Parametri	Tipo	Descrizione
Archivio	Espr_Archivio	Archivio contenente la selezione

Descrizione
SCHEDA PRECEDENTE imposta la scheda precedente a quella corrente dell'archivio *Archivio* come scheda corrente e la carica in memoria. Se nella selezione non esistono schede precedenti, la funzione Inizio selezione restituirà il valore Vero (funzione descritta in questo capitolo) e non avrete più alcuna scheda corrente, poichè vi trovate prima dell'inizio della selezione.
Se non esiste nessuna scheda corrente o nessuna selezione, SCHEDA PRECEDENTE non eseguirà alcuna azione. *Archivio* è opzionale. Se questo argomento viene omesso, SCHEDA PRECEDENTE si riferirà all'archivio corrente

Riferimenti
SCHEDA SUCCESSIVA, Inizio selezione

Fine selezione — End selection

Sintassi
Fine selezione({Archivio})
➤ Booleano

Parametri	Tipo	Descrizione
Archivio	Espr_Archivio	Archivio contenente la selezione da verificare

Descrizione
Fine selezione è una funzione che resituisce il valore Vero se, con il comando SCHEDA SUCCESSIVA, vi siete collocati oltre l'ultima scheda della selezione appartenente all'archivio *Archivio*. *Archivio* è opzionale. Se questo argomento viene omesso, Fine selezione si riferirà alla selezione corrente dell'archivio in uso.

Nota
Dopo la chiamata dei comandi STAMPA SELEZIONE, APPLICA ALLA SELEZIONE, la funzione Fine selezione restituirà il valore Vero.

Esempio
Fate riferimento all'esempio della descrizione della funzione Inizio selezione.

Riferimenti
SCHEDA SUCCESSIVA, Inizio selezione

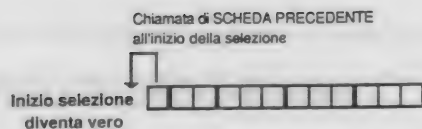
Inizio selezione — Before selection

Sintassi
Inizio selezione({Archivio})
➤ Booleano

Parametri	Tipo	Descrizione
Archivio	Espr_Archivio	Archivio contenente la selezione da verificare

Descrizione
Inizio selezione è una funzione che restituisce il valore Vero se, usando SCHEDA PRECEDENTE, vi siete collocati prima della prima scheda della selezione dell'archivio *Archivio*. *Archivio* è opzionale. Se questo argomento viene omesso, Inizio selezione si riferirà alla selezione corrente dell'archivio in uso.

Chiamata di SCHEDA PRECEDENTE all'inizio della selezione

Inizio selezione diventa vero

Esempio
Questa procedura generale accetta due valori di spostamento "Avanti" per passare alla scheda successiva e "Indietro" per passare a quella precedente. Se oltrepassate i limiti della selezione, ad esempio oltre l'ultima scheda, ritornerete alla prima.
Potete chiamarla scrivendo, ad esempio, *Spostati*("Avanti") o *Spostati*("Indietro")

```
Azione:=$1
Caso fra
    :(Azione="Avanti")
    SCHEDA SUCCESSIVA                     Vai alla scheda successiva
    Se(Fine selezione)                    Se vai troppo avanti
        PRIMA SCHEDA                      Ritorna alla prima scheda
    Fine se
    :(Azione="Indietro")
    SCHEDA PRECEDENTE                     Passa alla scheda precedente
    Se(Inizio selezione)                  Se vai troppo indietro
        ULTIMA SCHEDA                     Vai all'ultima scheda.
    Fine se
Fine caso
```

Riferimenti
SCHEDA PRECEDENTE, Fine selezione

Schede in selezione — Records in selection

Sintassi
Schede in selezione({Archivio})
➤ Intero Lungo

Parametri	Tipo	Description
Archivio	Espr_Archivio	Archivio contenente la selezione

Descrizione
Schede in selezione restituisce il numero di schede della selezione corrente dell'archivio *Archivio*. *Archivio* è opzionale. Se questo argomento viene omesso, Schede in selezione si riferirà alla selezione dell'archivio corrente.

Esempio

```
AVVISO("Ci sono "+Stringa(Schede in selezione)+" schede selezionate")
```

Riferimenti
Schede in archivio

SELEZIONA SCHEDA ATTUALE — ONE RECORD SELECT

Sintassi
SELEZIONA SCHEDA ATTUALE({Archivio})

Parametri	Tipo	Descrizione
Archivio	Espr_Archivio	Archivio contenente la selezione

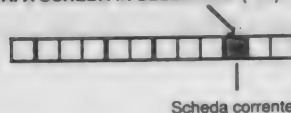
Descrizione
SELEZIONA SCHEDA ATTUALE crea una nuova selezione che comprende soltanto la scheda corrente della selezione di *Archivio*. Se non esiste nessuna scheda corrente, SELEZIONA SCHEDA ATTUALE non esegue alcuna azione. *Archivio* è opzionale. Se questo argomento viene omesso, SELEZIONA SCHEDA ATTUALE si riferirà all'archivio corrente.

Riferimenti
TUTTE LE SCHEDE

VAI A SCHEDA IN SELEZIONE — GOTO SELECTED RECORD

Sintassi
VAI A SCHEDA IN SELEZIONE({Archivio};Posizione)

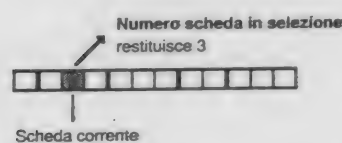
Parametri	Tipo	Descrizione
Archivio	Espr_Archivio	Archivio contenente la selezione
Posizione	Espr_NumPos	Numero della scheda relativo alla selezione

Descrizione
VAI A SCHEDA IN SELEZIONE imposta la scheda specificata come scheda corrente di *Archivio*. Se *Posizione* è superiore al numero di schede della selezione, VAI A SCHEDA IN SELEZIONE imposterà come scheda corrente l'ultima scheda della selezione. Se *Posizione* è inferiore a 1, VAI A SCHEDA IN SELEZIONE imposterà come scheda corrente la prima scheda della selezione. Se non c'è selezione corrente, VAI A SCHEDA IN SELEZIONE non esegue alcuna azione.

Riferimenti
Numero scheda nella selezione, Numero scheda, VAI A SCHEDA

Numero scheda in selezione | Selected record number

Sintassi
Numero scheda in selezione({Archivio})
➤ Intero Lungo Positivo

Parametri	Tipo	Descrizione
Archivio	Espr_Archivio	Archivio contenente la selezione

Descrizione
Numero scheda in selezione restituisce il numero della scheda corrente relativo alla selezione di *Archivio.*
- Se la scheda corrente è la prima della selezione, Numero scheda in selezione restituisce il valore1,
- Se la scheda corrente è l'ultima della selezione, Numero scheda in selezione restituisce il numero di schede in selezione, ossia il valore indicato da Schede in selezione
- Se Inizio selezione è Vero, Numero scheda in selezione restituirà il valore 0
- Se Fine selezione è Vero, Numero scheda in selezione restituirà il numero di schede -1.

Riferimenti
Numero scheda, VAI A SCHEDA IN SELEZIONE, PRIMA SCHEDA

VISUALIZZA SELEZIONE | DISPLAY SELECTION

Sintassi
VISUALIZZA SELEZIONE({Archivio};{*})

Parametri	Tipo	Descrizione
Archivio	Espr_Archivio	Archivio contenente la selezione da visualizzare
*	Costante	Permette di ottenere la visualizzazione in formato Lista

Descrizione
VISUALIZZA SELEZIONE presenta sullo schermo la lista delle schede della selezione dell'archivio *Archivio*, utilizzando il formato Lista corrente. Facendo doppio clic su una delle schede, questa vi viene presentata in formato Pagina senza la possibilità di modificarla. *Archivio* è opzionale. Se questo argomento viene omesso, VISUALIZZA SELEZIONE si riferirà all'archivio corrente.

Se la selezione di *Archivio* è vuota, un messaggio vi indicherà che la selezione è vuota.
Se viene selezionata una sola scheda, questa comparirà direttamente in formato Pagina. In tal caso, l'argomento opzionale * permetterà di ottenere la visualizzazione della scheda in formato Lista. In tutti i casi, se utilizzate l'argomento * facendo doppio clic su una scheda, il formato Pagina sarà visualizzato senza barra di scorrimento.

Cicli delle procedure e degli script:

[1] Prima chiamata di VISUALIZZA SELEZIONE
    Chiamata di Prima del formato Lista una volta
[2] Aggiornamento della lista
    Chiamata di In intestazione del formato Lista una volta
    Chiamata di Prima e Durante del formato Lista per ogni scheda
[3] Clic su un pulsante del formato Lista
    Chiamata di Durante del formato Lista
[4] Doppio clic su una scheda
    Chiamata di Durante del formato Lista
    Chiamata di Prima del formato Pagina
[5] Passaggio ad un'altra scheda
    Chiamata di Durante del formato Pagina della scheda corrente
    Chiamata di Prima del formato Pagina della nuova scheda corrente
[6] Clic su Conferma nel formato Pagina
    Chiamata di Durante del formato Pagina
    Chiamata dei cicli descritti in [2] Aggiornamento della lista
[7] Clic su Finito in formato Lista
    Chiamata di Durante del formato Lista

Pulsanti personalizzati

Se non ci sono pulsanti nel pié di pagina del formato, 4th DIMENSION visualizza il pulsante predeterminato **Finito**; altrimenti, potete inserire i vostri pulsanti nel pié di pagina del formato.

Potete uscire da VISUALIZZA SELEZIONE:
- facendo clic sul pulsante "Finito",
- facendo clic su un pulsante di conferma o di annullamento da voi creato
- chiamando uno dei comandi CONFERMA o ANNULLA

Durante VISUALIZZA SELEZIONE potete selezionare le schede per mezzo del mouse, come in ambiente Utilizzo.
- Maiuscole-Clic: estende la selezione alle schede comprese tra la scheda selezionata e quella su cui fate clic;
- Comando-Clic: aggiunge alla selezione la scheda su cui fate clic;
- il comando **Seleziona Tutto** del menu **Composizione** seleziona tutte le schede.
Quando uscite da VISUALIZZA SELEZIONE e alcune schede sono selezionate, queste vengono raccolte in un insieme chiamato "UserSet".

Nota

Non potete cambiare il formato Pagina o il formato Lista corrente durante l'esecuzione di VISUALIZZA SELEZIONE.

Esempio
Prendiamo l'esempio di un archivio [Animali] contenente 3 campi. Vogliamo presentare questo archivio sotto forma di lista con la possibilità di cancellare o stampare le schede selezionate. Il formato Lista sarà quindi il seguente:

Associate ad una scelta di menu la seguente procedura *ListaAnimali*:

```
IMPOSTA ARCHIVIO CORRENTE([Animali])
TUTTE LE SCHEDE
FORMATO LISTA("ListaAnimali")
Ripeti
    TUTTE LE SCHEDE
    VISUALIZZA SELEZIONE
    Caso fra
        :(Stampa=1)                   Se faccio clic sul pulsante "Stampa"
        USA INSIEME("UserSet")        Recupera l'insieme delle schede selezionate
        STAMPA SELEZIONE              E stampa le schede selezionate
        :(Cancella=1)                 Se faccio clic sul pulsante "Cancella"
        USA INSIEME("UserSet")        Recupera l'insieme delle schede selezionate
        CANCELLA SELEZIONE            E cancella le schede selezionate
    Fine caso
Finché(Esci=1)                        Ripeti l'operazione finché non faccio clic sul pulsante
                                      "Esci"
```

Un altro esempio: gestione di un doppio clic
Potete scoprire se se è stato fatto doppio clic su una scheda verificando l'esistenza contemporanea di queste 3 condizioni:
- **Durante** restituisce vero
- **Prima** restituisce falso
- Nessuno dei pulsanti in pié di pagina restituisce 1
In questo caso potete, ad esempio, modificare il formato Pagina in funzione di un'informazione contenuta nella scheda su cui avete fatto doppio clic.

Procedura associata al formato Lista

```
Se(Durante)&(Non(Prima))&(pEsci=0)                    Si tratta di un doppio clic
    Caso fra
        :([Archivio]Genere="Sig")                      Se il genere è Signore
        FORMATO PAGINA([Archivio];"Signore")           Seleziona il formato specifico dei signori
         ([Archivio]Genere="Sigra")                    Se il genere è Signora
        FORMATO PAGINA([Archivio];"Signora")           Seleziona il formato specifico delle signore
        :([Archivio]Genere=Signa")                     Se il genere è Signorina
        FORMATO PAGINA([Archivio]"Signorina")          Seleziona il formato specifico delle signorine
    Fine caso
Fine se
```

Riferimenti
MODIFICA SELEZIONE, USA INSIEME, FORMATO PAGINA, FORMATO LISTA

CANCELLA SELEZIONE — DELETE SELECTION

Sintassi
CANCELLA SELEZIONE({Archivio})

Parametri	**Tipo**	**Descrizione**
Archivio	Espr_Archivio	Archivio contenente la selezione da cancellare

Descrizione
CANCELLA SELEZIONE cancella le schede della selezione dell'archivio *Archivio*. Le schede così cancellate vengono irrimediabilmente perse: deve perciò essere richiesta la conferma della cancellazione. Dopo una chiamata di CANCELLA SELEZIONE, la selezione di *Archivio* è vuota. Durante la chiamata di CANCELLA SELEZIONE, non è possibile chiamare altre procedure, nè relazioni automatiche. Se volete effettuare aggiornamenti sulle schede cancellate, dovrete farlo prima della chiamata di CANCELLA SELEZIONE. *Archivio* è opzionale. Se questo argomento viene omesso, CANCELLA SELEZIONE si riferirà all'archivio corrente.

Nota relativa all'uso in Multi-utenza
Se state lavorando in Multi-utenza ed altri utenti stanno lavorando su alcune schede, 4th DIMENSION creerà un insieme chiamato "LockedSet" contenente le schede in uso che non è stato possibile cancellare.

Esempio

```
IMPOSTA ARCHIVIO CORRENTE([Abbonati])                         Cerca gli abbonati che non hanno pagato
                                                              la loro quota
CERCA([Abbonati]Quota=0)
Se(Schede in selezione>0)                                     Se trovi delle schede
    CONFERMA("Sei sicuro di voler cancellare questi abbonati?")   Chiedi di confermare la cancellazione
    Se(OK=1)                                                  Se la cancellazione è confermata
        CANCELLA SELEZIONE                                    Cancella la selezione degli abbonati
    Fine se
Fine se
```

Riferimenti
CANCELLA SCHEDA

APPLICA ALLA SELEZIONE — APPLY TO SELECTION

Sintassi
APPLICA ALLA SELEZIONE({Archivio};Istruzione)

Parametri	Tipo	Descrizione
Archivio	Espr_Archivio	Archivio contenente la selezione
Istruzione	Espr_Alfa	Istruzione da eseguire per ogni scheda della selezione

Descrizione
APPLICA ALLA SELEZIONE applica un'istruzione ad ogni scheda della selezione dell'archivio *Archivio*. Una finestra contenente un indicatore di progressione visualizza lo svolgimento dell'operazione.
APPLICA ALLA SELEZIONE è un'operazione sequenziale che agisce sulle schede una alla volta. Non potete interrompere l'operazione da programma, mentre invece potete interrompere l'esecuzione facendo clic sul pulsante Fine. Potete impedire la visualizzazione dell'indicatore di progressione con il comando DISATTIVA MESSAGGI prima della chiamata di APPLICA ALLA SELEZIONE.
Se la procedura modifica il valore di uno dei campi dell'archivio *Archivio*, le schede verranno automaticamente registrate. Altrimenti, le schede verranno semplicemente rilette in modo sequenziale.
Una volta eseguita l'istruzione potrete sapere se l'operazione è stata completata verificando la variabile OK, che assumerà allora il valore 1; assumerà invece il valore 0 se avete interrotto l'operazione premendo il pulsante Fine o se l'operazione è stata interrotta per cause indipendenti dalla vostra volontà (disco pieno,...). *Archivio* è opzionale. Se viene omesso, APPLICA ALLA SELEZIONE agirà sull'archivio corrente.

L'istruzione APPLICA ALLA SELEZIONE non è soltanto destinata a modificare una selezione di schede. Potete anche utilizzarla per rileggere i valori di una selezione di schede, ed è più rapida rispetto ad un ciclo del tipo:

```
Mentre(Non(Fine selezione))
    Istruzione
    SCHEDA SUCCESSIVA
Fine Mentre
```

Questo ciclo vi offre il vantaggio di poter interrompere l'operazione tramite programmazione e far scorrere la selezione nei due sensi, mentre APPLICA ALLA SELEZIONE parte sempre dalla prima scheda.

Nota relativa all'uso in Multi-utenza
Se state lavorando in Multi-utenza e altri utenti stanno lavorando su alcune schede della selezione, 4th DIMENSION creerà un insieme chiamato "LockedSet", che conterrà la schede in uso non aggiornate dalla vostra procedura.

Esempio
Aumento del 5% dei salari dell'archivio [Persone]:
APPLICA ALLA SELEZIONE([Persone];[Persone]Salario:=[Persone]Salario*1,05)

Ogni anno, vogliamo aggiornare l'archivio clienti incrementandone l' età di un anno e riportando a zero il fatturato totale dei loro acquisti.
Non possiamo scrivere questa procedura su una sola riga, dobbiamo creare una procedura chiamata *NuovoAnno.*

[Clienti]Età:=[Clienti]Età+1
[Clienti]Fatturato:=0

IMPOSTA ARCHIVIO CORRENTE([Clienti])
TUTTE LE SCHEDE
APPLICA ALLA SELEZIONE(NuovoAnno) Aggiornamento della selezione clienti.

Riferimenti
MODIFICA SELEZIONE

MAILING SU SELEZIONE — MERGE SELECTION

Sintassi
MAILING SU SELEZIONE({Archivio};{NomeDoc})

Parametri	Tipo	Descrizione
Archivio	Espr_Archivio	Archivio su cui eseguire il mailing
NomeDoc	Espr_Alfa	Nome del documento Macintosh

Descrizione
MAILING SU SELEZIONE permette di unire i dati ad un documento esterno. Generalmente, si tratta di un documento creato con un elaboratore di testi, per eseguire un mailing. MAILING SU SELEZIONE realizza l'unione fra il documento tipo indicato con NomeDoc e i dati appartenenti alla selezione delle schede di *Archivio.* Il comando MAILING SU SELEZIONE necessita dell'installazione di un "merge module". Per ulteriori informazioni sugli elaboratori di testi di cui è disponibile il modulo rivolgetevi al vostro rivenditore di fiducia.

Sotto strutture

NUOVA SOTTO_SCHEDA — CREATE SUBRECORD

Sintassi
NUOVA SOTTO_SCHEDA(IlSottoarchivio)

Parametri	Tipo	Descrizione
IlSottoarchivio	Espr_SottoArch	Sotto-archivio in cui viene creata la sotto-scheda

Descrizione
NUOVA SOTTO_SCHEDA crea una sotto-scheda nel sotto-archivio *IlSottoarchivio* impostandola come sotto-scheda corrente, che verrà registrata soltanto se la scheda a cui appartiene è registrata. Se non c'è nessuna scheda corrente nell'archivio Il *Sottoarchivio*, NUOVA SOTTO_SCHEDA non esegue alcuna azione. Se desiderate proporre una sotto-scheda per l'inserimento, usate il comando AGGIUNGI SOTTO_SCHEDA.
IlSottoarchivio deve essere sempre preceduto dal nome dell'archivio a cui appartiene.

Esempio
Script del pulsante AggiungiLingua

```
x:=Richiedi("Lingua parlata?")             Richiedi la lingua corrente da aggiungere
Se(OK=1)
CREA SOTTO_SCHEDA([Persone]Lingua)         Crea una sotto-scheda
[Persone]Lingua'Nome=x                     Assegna il valore nel sotto-campo
Fine se
```

Riferimenti
AGGIUNGI SOTTO_SCHEDA

CANCELLA SOTTO_SCHEDA — DELETE SUBRECORD

Sintassi
CANCELLA SOTTO_SCHEDA(IlSottoarchivio)

Parametri	Tipo	Descrizione
IlSottoarchivio	Espr_SottoArch	Sotto-archivio in cui viene cancellata la sotto-scheda

Descrizione
CANCELLA SOTTO_SCHEDA cancella la sotto-scheda corrente di *IlSottoarchivio*. Se non esiste alcuna sotto-scheda corrente, CANCELLA SOTTO_SCHEDA non esegue alcuna azione. Dopo una chiamata di CANCELLA SOTTO_SCHEDA, la sotto-selezione di *IlSottoarchivio* è vuota. CANCELLA SOTTO_SCHEDA non può quindi essere usato per cancellare una selezione di sotto-schede.

La cancellazione di una sotto-scheda diventa definitiva soltanto quando è stata effettuata la registrazione della scheda a cui appartiene *IlSottoarchivio* . La cancellazione della scheda a cui appartiene *IlSottoarchivio* comporta evidentemente la cancellazione di tutte le sotto-schede di *IlSottoarchivio.*

Esempio
Ricerca della sotto-scheda [Risultati]Mese, il cui sotto-campo IlMese è uguale a Gennaio.

```
CERCA SOTTO_SCHEDE([Risultati]Mese'IlMese="Gennaio")
CANCELLA SOTTO_SCHEDE([Risultati]IlMese)
```

Non esistono comandi in 4th DIMENSION che permettano di cancellare una sotto-selezione.
Ecco come realizzare questa operazione:

```
TUTTE LE SOTTO_SCHEDE([Risultati]Mese)                  Seleziona tutte le sotto-schede
Mentre(Sotto_schede selezionate([Risultati]Mese>0))     Finché ci saranno delle sotto-schede
   CANCELLA SOTTO_SCHEDA([Risultati]Mese)               Elimina la sotto-scheda corrente
   TUTTE LE SOTTO_SCHEDE([Risultati]Mese)               Riprendi tutte le sotto-schede rimaste
Fine mentre
```

Riferimenti
CERCA SOTTO_SCHEDE, TUTTE LE SOTTO_SCHEDE

TUTTE LE SOTTO_SCHEDE — ALL SUBRECORDS

Sintassi
TUTTE LE SOTTO_SCHEDE(IlSottoarchivio)

Parametri	Tipo	Descrizione
IlSottoarchivio	Espr_SottoArch	Sotto-archivio contenente le sotto-schede da selezionare

Descrizione
TUTTE LE SOTTO_SCHEDE seleziona tutte le sotto-schede di *IlSottoarchivio* e ne fa la sotto-selezione corrente. Se non c'è nessuna scheda corrente nell'archivio *IlSottoarchivio*, TUTTE LE SOTTO_SCHEDE non esegue alcuna azione. Quando una scheda contenente delle sotto-schede viene caricata in memoria, la sotto-selezione conterrà tutte le schede.
Attenzione: TUTTE LE SOTTO_SCHEDE non seleziona tutte le sotto-schede di una selezione di schede, la sua azione è limitata localmente ad una scheda e ad un livello di sotto-archivio.

Esempio
Fate riferimento all'esempio precedente

Riferimenti
CERCA SOTTO_SCHEDE, Sotto_schede selezionate

Sotto_schede selezionate — Records in subselection

Sintassi
Sotto_schede selezionate(IlSottoarchivio)
➤ Intero positivo

Parametri	Tipo	Descrizione
IlSottoarchivio	Espr_SottoArch	Sotto-archivio di cui volete conoscere il numero di elementi nella sotto-selezione

Descrizione
Sotto_schede selezionate restituisce il numero di sotto-schede contenute nella sotto-selezione del sotto-archivio *IlSottoarchivio*. Sotto_schede selezionate si riferisce alla scheda corrente dell'archivio a cui appartiene *IlSottoarchivio*, se non esist, Sotto_schede selezionate restituirà il valore 0.

Esempio

```
CERCA SOTTO_SCHEDE([Vendite]Mese'Vendite>1000)          Cerca le vendite superiori a 1000
Se(Sotto_schede selezionate([Vendite]Mese)>0)           Se il numero di sotto-schede selezionate è >0
    AVVISO(Stringa(Sotto_schede selezionate             Visualizza il numero delle vendite > 1000 per il
                  ([Vendite]Mese))+" al Mese")          mese
Fine se
```

Riferimenti
CERCA SOTTO_SCHEDE, TUTTE LE SOTTO_SCHEDE

PRIMA SOTTO_SCHEDA — FIRST SUBRECORD

Sintassi
PRIMA SOTTO_SCHEDA(IlSottoarchivio)

Parametri	Tipo	Descrizione
Il Sottoarchivio	Espr_SottoArch	Sotto-archivio

Descrizione
PRIMA SOTTO_SCHEDA seleziona la prima sotto-scheda della sotto-selezione corrente di *IlSottoarchivio*, facendola diventare la sotto-scheda corrente. Se la sotto-selezione è vuota o se non esiste nessuna scheda corrente nell'archivio a cui appartiene *IlSottoarchivio*, PRIMA SOTTO_SCHEDA non esegue alcuna azione. I comandi CERCA SOTTO_SCHEDE, ORDINA SOTTO_SCHEDE passano automaticamente alla prima sotto-scheda, perciò non è necessario chiamare PRIMA SOTTO_SCHEDA dopo la chiamata di uno di questi comandi.

Riferimenti
ULTIMA SOTTO_SCHEDA.

SOTTO_SCHEDA SUCCESSIVA — NEXT SUBRECORD

Sintassi
SOTTO_SCHEDA SUCCESSIVA(IlSottoarchivio)

Parametri	Tipo	Descrizione
IlSottoarchivio	Espr_SottoArch	Sotto-archivio

Descrizione
SOTTO_SCHEDA SUCCESSIVA seleziona la sotto-scheda successiva alla sotto-scheda corrente di *IlSottoarchivio*, facendola diventare la sotto-scheda corrente. Se SOTTO_SCHEDA SUCCESSIVA supera il limite della sotto-selezione, la funzione Fine sotto_selezione restituirà un valore vero. Se la sotto-selezione è vuota o se Inizio sotto-selezione restituisce il valore vero, SOTTO_SCHEDA SUCCESSIVA non esegue alcuna azione.

Esempio

```
Mentre(Non(Finesotto_selezione([Risultato]Mese)))      Finché non giungi alla fine della sotto-selezione
    [Risultato]Mese'CA:=0                              Azzera il campo CA
    SOTTO_SCHEDA SUCCESSIVA                            Passa alla sotto-scheda successiva
    ([Risultato]Mese)                                  Fine della sequenza
Fine mentre
```

Riferimenti
SOTTO_SCHEDA PRECEDENTE, Fine sotto_selezione

SOTTO_SCHEDA PRECEDENTE — PREVIOUS SUBRECORD

Sintassi
SOTTO_SCHEDA PRECEDENTE(IlSottoarchivio)

Parametri	Tipo	Descrizione
IlSottoarchivio	Espr_SottoArch	Sotto-archivio

Descrizione
SOTTO_SCHEDA PRECEDENTE seleziona la sotto-scheda che precede la sotto-scheda corrente di *IlSottoarchivio*, facendola diventare la sotto-scheda corrente. Se SOTTO_SCHEDA PRECEDENTE supera il limite della sotto-selezione, Inizio sotto_selezione restituirà il valore vero. Se la sotto-selezione è vuota o se la funzione Fine sotto-selezione restituisce il valore vero, SOTTO_SCHEDA PRECEDENTE non esegue alcuna azione.

Riferimenti
SOTTO_SCHEDA SUCCESSIVA, Inizio sotto_selezione

ULTIMA SOTTO_SCHEDA — LAST SUBRECORD

Sintassi
ULTIMA SOTTO_SCHEDA(IlSottoarchivio)

Parametri	Tipo	Descrizione
IlSottoarchivio	Espr_SottoArch	Sotto-archivio

Descrizione
ULTIMA SOTTO_SCHEDA seleziona l'ultima sotto-scheda della sotto-selezione corrente di *IlSottoarchivio*, facendola diventare la sotto-scheda corrente. Se la sotto-selezione è vuota o se non esiste nessuna scheda corrente nell'archivio a cui appartiene *IlSottoarchivio*, ULTIMA SOTTO_SCHEDA non esegue alcuna azione.

Riferimenti
PRIMA SOTTO_SCHEDA

Inizio sotto_selezione — Before subselection

Sintassi
Inizio sotto_selezione(IlSottoarchivio)
➤ Booleano

Parametri	Tipo	Descrizione
IlSottoarchivio	Espr_SottoArch	Sotto-archivio

Descrizione
Inizio sotto_selezione restituisce il valore vero quando la sotto-scheda corrente si trova prima della prima sotto-scheda di *IlSottoarchivio*. Accade quando chiamate il comando SOTTO_SCHEDA PRECEDENTE mentre vi trovate sulla prima sotto-scheda della sotto-selezione. Se la sotto-selezione è vuota o se non esiste nessuna scheda corrente, Inizio sotto_selezione restituirà il valore vero.

Riferimenti
Fine sotto_selezione, SOTTO_SCHEDA PRECEDENTE

Fine sotto_selezione — End subselection

Sintassi
Fine sotto_selezione(IlSottoarchivio)
➤ Booleano

Parametri	Tipo	Descrizione
IlSottoarchivio	Espr_SottoArch	Sotto-archivio

Descrizione
Fine sotto_selezione restituisce il valore vero quando la sotto-scheda corrente si trova dopo l'ultima sotto-scheda di *IlSottoarchivio.* Accade quando chiamate il comando SOTTO_SCHEDA SUCCESSIVA mentre vi trovate sull'ultima sotto-scheda della sotto-selezione. Se la sotto-selezione è vuota o se non esiste nessuna scheda corrente, Fine sotto_selezione restituirà il valore vero.

Riferimenti
Inizio sotto_selezione, SCHEDA PRECEDENTE

CERCA SOTTO_SCHEDE — SEARCH SUBRECORD

Sintassi
CERCA SOTTO_SCHEDE(IlSottoarchivio;Formula)

Parametri	Tipo	Descrizione
IlSottoarchivio	Espr_SottoArch	Sotto-archivio in cui verrà eseguita la ricerca
Formula	Espr_Bool	Formula di ricerca

Descrizione
CERCA SOTTO_SCHEDE effettua una ricerca nel sotto-archivio *IlSottoarchivio.* CERCA SOTTO_SCHEDE calcola il valore di *Formula* per ogni sottoscheda, e se *Formula* restituisce il valore vero, la sotto-scheda viene aggiunta alla sotto-selezione. Questa ricerca viene effettuata solo per le sotto-schede della scheda corrente dell'archivio a cui appartiene *IlSottoarchivio.* Se la sotto-selezione ottenuta non è vuota, la prima sotto-scheda di questa selezione diventa la sotto-scheda corrente. Se non esiste alcuna scheda corrente nell'archivio in uso o nessuna sotto-scheda corrente di livello superiore nel caso di sotto-strutture a più livelli, CERCA SOTTO_SCHEDE non esegue alcuna azione. Se l'archivio non presenta sotto-strutture, 4th DIMENSION vi segnala che non è in grado di eseguire l'istruzione.

Esempio
Script del pulsante pCanc, che permette la cancellazione di una Parola Chiave nel sotto-archivio Parole.

vChiave:=**Richiedi**("Parola Chiave da cancellare")	Richiedi la parola da cancellare
Se(OK=1)	Se la richiesta è confermata
CERCA SOTTO_SCHEDE(Parole;Parole'ParolaChiave=vChiave)	Cerca la parola nel sotto-arch. Parole
Se(**Sotto_schede selezionate**(Parole)=1)	Se trovi una sotto-scheda
CANCELLA SOTTO_SCHEDE(Parole)	Cancella la parola dal sotto-archivio Parole
Fine se	
Fine se	

Riferimenti
TUTTE LE SCHEDE, SOTTO_SCHEDE selezionate

ORDINA SOTTO_SELEZIONE — SORT SUBSELECTION

Sintassi
ORDINA SOTTO_SELEZIONE(IlSottoarch;IlSottoCampo1;{Ord1};...;{IlSottoCampo8;{Ord8}})

Parametri	Tipo	Descrizione
IlSottoarch	Espr_SottoArch	Sotto-archivio da ordinare
IlSottoCampo1...8	Espr_Campo	Campo di ordinamento
Ord1...8	> o <	Sequenza di ordinamento

Descrizione

ORDINA SOTTO_SELEZIONE ordina solo la sotto-selezione della scheda corrente dell'archivio a cui appartiene *IlSottoarc*. Se *Ordine* è uguale a ">", l'ordinamento sarà crescente; sarà invece decrescente se *Ordine* è uguale a "<". Al termine dell'ordinamento, 4th DIMENSION trasformerà la prima sotto-scheda in sotto-scheda corrente. L'ordinamento delle sotto-schede è un'operazione dinamica che non verrà registrata, anche se registrate la scheda a cui appartiene *IlSottoarc*. Se non esiste nessuna scheda corrente nell'archivio in uso o nessuna sotto-scheda corrente di livello superiore nel caso di sotto-strutture a più livelli, ORDINA SOTTO_SELEZIONE non esegue alcuna azione.

Esempio

Se(Prima)	Prima che il formato compaia sullo schermo
TUTTE LE SOTTO_SCHEDE(Par)	Seleziona tutte le sotto-schede
ORDINA SOTTO_SELEZIONE(Par;Par'Tipo;>;Par'ParCh;>)	Ordina le sotto-schede per tipo e per ParCh
Fine se	

Riferimenti

CERCA SOTTO_SCHEDE

APPLICA A SOTTO_SELEZIONE — APPLY TO SUBSELECTION

Sintassi

APPLICA A SOTTO_SELEZIONE(IlSottoarch;L'Istruzione)

Parametri	Tipo	Descrizione
IlSottoarch	Espr_SottoArch	Sotto-archivio da modificare
Istruzione	Espr_Alfa	Istruzione da applicare alla sotto-selezione

Descrizione

APPLICA A SOTTO_SELEZIONE esegue *Istruzione* per tutte le sotto-schede della sotto-selezione di *IlSottoarch*. *Istruzione* può essere un'istruzione comune o la chiamata di una procedura globale. Se la sotto-selezione di *IlSottoarch* è vuota o se non esiste alcuna scheda corrente nell'archivio a cui appartiene *IlSottoarch*, APPLICA A SOTTO_SELEZIONE non esegue alcuna azione. La modifica della sotto-selezione sarà registrata solo se la scheda a cui appartiene *IlSottoarch* verrà registrata.

Esempio

TUTTE LE SOTTO_SCHEDE(Righe)	Seleziona tutte le sotto-schede
APPLICA A SOTTO_SELEZIONE(Righe;Righe'IVA:=19)	Il campo IVA assume il valore 19 per tutte le sotto-schede

Riferimenti

TUTTE LE SOTTO_SCHEDE, CERCA SOTTO_SCHEDE

Cicli di esecuzione

Prima — Before

Sintassi
Prima
➤ Booleano

Contesto
Procedure di formato, procedure di archivio o script.

Descrizione
Prima è una funzione che restituisce il valore vero se l'esecuzione delle procedure si trova nel ciclo di Prima. Tale ciclo viene chiamato una volta prima della comparsa di un formato sullo schermo, dopo una chiamata di AGGIUNGI SCHEDA, MODIFICA SCHEDA o DIALOGO. Nel caso di una visualizzazione a schermo o di una stampa, il ciclo viene chiamato una volta per ogni scheda visualizzata o stampata. Fate riferimento all'appendice della prima parte di questo Manuale.
Il ciclo di Prima viene chiamato soltanto una volta prima di passare in inserimento: vi permetterà d'inizializzare alcuni parametri o campi prima di iniziare l'inserimento stesso.

Esempio

```
Se(Prima)                                   Prima che la scheda compaia sullo schermo
    Se(Data Creazione=!00-00-00!)           Prima della comparsa del formato, se Data Creazione è nulla
        Data Creazione:=Data odierna        Aggiorna la data di creazione
    Fine se
    Data Modif:=Data odierna                Aggiorna la data di modifica
Fine se
```

Riferimenti
Durante, Dopo

Durante — During

Sintassi
Durante
➤ Booleano

Contesto
Procedure di formato, procedure di archivio o script.

Descrizione
Durante è una funzione che restituisce il valore vero se l'esecuzione delle procedure si trova nel ciclo di Durante.
Questo ciclo viene chiamato:
- quando avete modificato un campo o una variabile e passate ad un altro campo;
- quando fate clic su un pulsante o su un pulsante d'opzione;
- quando fate clic su un'area esterna;
- quando scegliete un comando da menu, ad eccezione dei menu Mela, Composizione e dei menu associati al formato;
- quando fate clic su un'area di scorrimento , su un menu a comparsa, su un indicatore di progressione;
- quando modificate una sotto-scheda o una scheda di un archivio incluso, e soltanto per esse;
- quando una sotto-scheda o una scheda di un archivio incluso devono essere aggiornate sullo schermo.

Inoltre, questo ciclo viene chiamato un'ultima volta al momento della conferma della scheda. Nel caso di una visualizzazione a schermo o di una stampa, il ciclo verrà chiamato una volta per ogni scheda visualizzata o stampata. Fate riferimento all'appendice della prima parte di questo Manuale.

Esempio

```
Se(Durante)                                                   Se Durante è vero
    [Fatture]TotaleLordo:=[Fatture]TotaleNetto+[Fatture]IVA   Ricalcola l'importo lordo della fattura
Fine se
```

Riferimenti
Prima, Dopo

Dopo — After

Sintassi
Dopo
➤ Booleano

Contesto
Procedure di formato, procedure di archivio o script.

Descrizione
Dopo è una funzione che restituisce il valore vero se l'esecuzione delle procedure si trova nel ciclo di Dopo. Questo ciclo viene chiamato soltanto una volta e unicamente se la scheda in inserimento è stata confermata; non viene mai chiamato nel corso di una stampa o di una visualizzazione.
Dal momento che questo ciclo viene chiamato soltanto una volta ed unicamente se la scheda è stata confermata, potete effettuare aggiornamenti su relazione, ad esempio, l'aggiornamento del totale di un fatturato.
Per informazioni più dettagliate sui cicli di esecuzione fate riferimento all'appendice della prima parte di questo Manuale.

Esempio

Se(Dopo)
    [Clienti]Fatturato:=[Clienti]Fatturato+Totale fattura
Fine se

Immaginiamo una struttura contenente due archivi, Fatture e Clienti, collegati per mezzo di una relazione automatica.
Se viene confermata una nuova fattura, dobbiamo aggiornare il fatturato.

Riferimenti
Prima, Durante

In intestazione — In header

Sintassi
In intestazione
➤ Booleano

Contesto
Procedura di formato durante una stampa, visualizzazione su schermo.

Descrizione
In intestazione è una funzione che restituisce il valore vero durante la stampa dell'intestazione di un formato o durante una chiamata di VISUALIZZA/MODIFICA SELEZIONE.

In intestazione è un ciclo di esecuzione a tutti gli effetti, non deve essere inserito in Prima o in Durante, perché non verrà eseguito.
Al momento della stampa, potete sapere se 4th DIMENSION sta stampando la prima intestazione per mezzo della funzione Inizio selezione.
Per informazioni più dettagliate sui cicli di esecuzione fate riferimento all'appendice della prima parte di questo Manuale.

Esempio

Se(In intestazione)&(Inizio selezione)
vtitolo:="Rapporto vendite 89"
Fine se

Inizializzazione della variabile titolo nella prima intestazione.

Riferimenti

In rottura, In pié di pagina

In rottura

In break

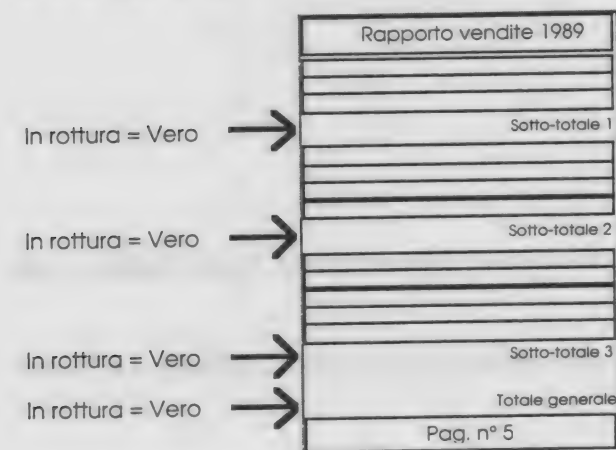
Sintassi

In rottura
➤ Booleano

Contesto

Stampa

Descrizione

In rottura è una funzione che restituisce il valore vero durante la stampa delle zone di rottura di un formato, e ad ogni livello di rottura.

In rottura è un ciclo di esecuzione a tutti gli effetti. Non dev'essere introdotto in Prima o Durante inserimento, poichè non verrà eseguito.

Esempio

Se(In rottura)
v:=**Sottototale**([Fatture]ImportoNetto)
Fine se

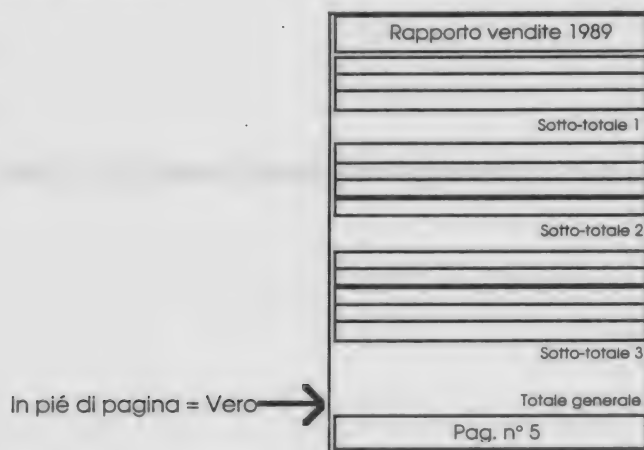
Riferimenti

In intestazione, In pié di pagina

In pié di pagina   In footer

Sintassi
In pié di pagina
➤ Booleano

Contesto
Procedura di formato durante una stampa

Descrizione
In pié di pagina è una funzione che restituisce il valore vero durante la stampa della zona in pié di pagina di un formato.

Potete verificare se state stampando l'ultima pagina servendovi della funzione Fine selezione, che restituirà il valore vero soltanto in questo caso.

In pié di pagina è un ciclo di esecuzione a tutti gli effetti. Non deve essere introdotto in Prima o in Durante, perchè in questo caso non verrà eseguito.

Esempio

```
Se(In pié di pagina)
    vPagina:=Pagina in stampa
Fine se
```

Stampa il numero di pagina in piè di pagina
vPagina assume il valore del numero di pagina in stampa

```
Se(In piè di pagina)&(Fine selezione)
    vtitolo:="Fine rapporto"
Fine se
```

Se giungi al pié di pagina del formato e se Fine selezione restituisce il valore vero, si tratta dell'ultima pagina del rapporto.

Riferimenti
In intestazione, In rottura.

PRIMA PAGINA — FIRST PAGE

Sintassi
PRIMA PAGINA

Contesto
Inserimento in un formato

Descrizione
PRIMA PAGINA cambia la pagina corrente di un formato per visualizzarne la prima pagina. Se questa è già visualizzata, PRIMA PAGINA non esegue alcuna azione.

Riferimenti
ULTIMA PAGINA, PAGINA SUCCESSIVA, PAGINA PRECEDENTE, VAI A PAGINA, Pagina attuale

ULTIMA PAGINA — LAST PAGE

Sintassi
ULTIMA PAGINA

Contesto
Inserimento in un formato

Descrizione
ULTIMA PAGINA cambia la pagina corrente di un formato per visualizzarne l'ultima pagina. Se questa è già visualizzata, ULTIMA PAGINA non esegue alcuna azione.

Riferimenti
PRIMA PAGINA, PAGINA SUCCESSIVA, PAGINA PRECEDENTE, VAI A PAGINA, Pagina attuale

PAGINA PRECEDENTE — PREVIOUS PAGE

Sintassi
PAGINA PRECEDENTE

Contesto
Inserimento in un formato

Descrizione
PAGINA PRECEDENTE cambia la pagina corrente di un formato per visualizzarne la pagina precedente. Se la pagina visualizzata è la prima del formato, PAGINA PRECEDENTE non esegue alcuna azione.

Riferimenti
PRIMA PAGINA, ULTIMA PAGINA, PAGINA SUCCESSIVA, VAI A PAGINA, Pagina attuale

---

PAGINA SUCCESSIVA — NEXT PAGE

Sintassi
PAGINA SUCCESSIVA

Contesto
Inserimento in un formato

Descrizione
PAGINA SUCCESSIVA cambia la pagina corrente di un formato per visualizzare la pagina successiva. Se la pagina visualizzata è l'ultima del formato, PAGINA SUCCESSIVA non esegue nessuna azione.

Riferimenti
PRIMA PAGINA, ULTIMA PAGINA, PAGINA PRECEDENTE, VAI A PAGINA, Pagina attuale

---

VAI A PAGINA — GOTO PAGE

Sintassi
VAI A PAGINA(NrDiPagina)

Parametri	Tipo	Descrizione
NrDiPagina	Espr_NumPos	Numero della pagina da visualizzare

Contesto
Inserimento in un formato

Descrizione
VAI A PAGINA cambia la pagina corrente di un formato per visualizzare la pagina indicata con il *NrDiPagina*. Se il numero della pagina richiesta è inferiore a 1, VAI A PAGINA passa alla prima pagina; se è superiore al numero di pagine del formato, VAI A PAGINA passa all'ultima pagina.

Riferimenti
PRIMA PAGINA, ULTIMA PAGINA, PAGINA SUCCESSIVA, PAGINA PRECEDENTE, Pagina attuale

Pagina attuale

Sintassi
Pagina attuale
➤ Intero positivo

Contesto
Inserimento in un formato

Descrizione
Pagina attuale fornisce il numero di pagina del formato in inserimento.

Esempio
Potete, ad esempio, servirvi di questo comando per stabilire quali procedure di formato volete eseguire in base alla pagina visualizzata.

Se(**Durante**)	
Caso fra	
:(**Pagina attuale**=1)	Se la pagina corrente è pagina1, chiama la procedura *PF_Pagina1*
PF_Pagina1	
:(**Pagina attuale**=2)	Se la pagina corrente è pagina2, chiama la procedura *PF_Pagina2*
PF_Pagina2	
:(**Pagina attuale**=3)	Se la pagina corrente è pagina3, chiama la procedura *PF_Pagina3*
PF_Pagina3	
Fine caso	
Fine se	

Riferimenti
PRIMA PAGINA, ULTIMA PAGINA, PAGINA SUCCESSIVA, PAGINA PRECEDENTE, VAI A PAGINA

CONFERMA
CONFIRM

Sintassi
CONFERMA

Contesto
Inserimento, Dialogo, Visualizzazione su schermo

Descrizione
CONFERMA agisce come quando fate clic su un pulsante di conferma in un formato di inserimento o in una finestra di dialogo. La variabile di sistema OK assume il valore 1.Se CONFERMA viene chiamato nelle procedure di formato di un formato di tipo Lista usato con MODIFICA SELEZIONE o VISUALIZZA SELEZIONE, la chiamata di CONFERMA equivale a fare clic sul pulsante in pié di pagina del formato.

Note
1. La chiamata di CONFERMA dà luogo alla conferma della scheda, ma vi fa uscire dall'inserimento soltanto al termine di tutti i cicli, eseguiti con le rispettive istruzioni. In altri termini, la scheda non viene immediatamente registrata dopo la chiamata di CONFERMA.
2. Fate riferimento alla sezione Multi-utenza ed in particolare al comando INIZIA TRANSAZIONE, per la chiamata di CONFERMA durante la transazione.

Riferimenti
ANNULLA

ANNULLA
CANCEL

Sintassi
ANNULLA

Contesto
Inserimento, Dialogo, Visualizzazione su schermo

Descrizione
ANNULLA agisce come quando fate clic sul pulsante di annullamento in un formato di inserimento o in una finestra di dialogo. La variabile di sistema OK assume il valore 0. Se ANNULLA viene chiamato nelle procedure di formato di un formato di tipo Lista usato con MODIFICA SELEZIONE o VISUALIZZA SELEZIONE, la chiamata di ANNULLA equivale a fare clic sul pulsante in pié di pagina del formato.

Riferimenti
CONFERMA

BEEP BEEP

Sintassi
BEEP({Durata})

Parametri	Tipo	Descrizione
Durata	Espr_Num	Durata del beep

Descrizione
BEEP emette un beep attraverso l'altoparlante del Macintosh. Se *Durata* viene omesso, il beep dura mezzo secondo, altrimenti BEEP emetterà un suono pari alla *Durata* stabilita. (La durata è espressa in tick; un tick equivale a 1/60° di secondo). Se la versione del vostro System è 6.0 o successive, BEEP emetterà il beep di sistema scelto nel pannello di controllo. In questo caso, l'argomento *Durata* sarà superfluo. Se desiderate l'emissione di un suono particolare, usate il comando SUONA descritto qui di seguito.

Esempio

BEEP(60)	Emette un beep per un secondo

Riferimenti
SUONA

SUONA PLAY

Sintassi
SUONA(NomeSuono;{Canale})

Parametri	Tipo	Descrizione
NomeSuono	Espr_Alfa	Nome di una risorsa di tipo 'snd '
Canale	Espr_NumPos	Canale di uscita del suono (opzionale)

Descrizione
SUONA vi permette di ottenere un suono registrato sotto forma di risorsa di tipo 'snd ', chiamato *NomeSuono*. Tale suono deve trovarsi in uno degli archivi di risorse aperti, ossia il documento *NomeBaseDati*, il documento Proc.Ext o nel System.

Se *Canale* viene omesso, *NomeSuono* viene ottenuto in sincronia; questo significa che l'esecuzione delle procedure di 4th DIMENSION verrà interrotta finchè il suono non sarà completamente terminato. Se passate un valore in *Canale*, il suono viene eseguito in modo asincrono, ossia suonerà senza interrompere l'esecuzione delle vostre procedure. Il numero del canale da passare dipende dal suono che utilizzerete.

Canale può assumere i seguenti valori:
0: canale configurato automaticamente dal Sistema. Vi consigliamo di utilizzare questo canale quando non conoscete il tipo di suono che volete usare.
1 e 3: suoni semplici
5: suono digitalizzato
7 e 9: canali MIDI

Nel caso di chiamate asincrone, se volete ottenere suoni anche da altre applicazioni, dovrete liberare il canale suono scrivendo:

SUONA("";NumCan)

Esempio

Se(PulsAiuto=1)	
SUONA("Tarzan")	Se viene fatto clic sul pulsante di Aiuto
ApriAiuto	Avverti che qualcuno ha sentito la richiesta di aiuto
Fine se	Chiama la procedura di Aiuto

Riferimenti
BEEP

RIFIUTA — REJECT

SINTASSI
RIFIUTA({IlCampo})

Parametri	**Tipo**	**Descrizione**
IlCampo	Espr_Campo	Campo in cui avviene il rifiuto

Contesto
Durante dell'inserimento delle procedure di formato, delle procedure di archivio e degli script.

Descrizione
Questo comando vi permette di rifiutare l'inserimento del campo *IlCampo* appena modificato. Per non bloccare all'infinito l'inserimento di questo campo, è importante chiamare questo comando soltanto dopo un test sul valore inserito.
Se *IlCampo* viene omesso, RIFIUTA si riferirà all'inserimento globale di una scheda. Chiamando RIFIUTA in Durante, ogni tentativo di conferma verrà annullato. Come per la prima sintassi di RIFIUTA, dovete chiamare questa procedura soltanto dopo un test, per non bloccare all'infinito la conferma della scheda.

L'utilizzazione di questa procedura ha senso unicamente nel ciclo di Durante delle **procedure di archivio, delle procedure di formato o degli script**, in quanto è associata all'inserimento di una scheda o di un campo.

Esempio

Se(Salario>2000000)	Se il salario è superiore a 2000000
AVVISO("Il salario è troppo elevato!")	Avverti che l'inserimento è rifiutato
RIFIUTA(Salario)	Rifiuta l'inserimento
Fine se	

Esempio di rifiuto completo di una scheda:

Se(**Durante**)	
Se(Salario=0|(Nome="")	Se il salario o il nome non sono stati inseriti
RIFIUTA	Rifiuta la conferma della scheda
Fine se	
Fine se	

Riferimenti
VAI SU AREA

VAI SU AREA — GOTO AREA

Sintassi
VAI SU AREA(Oggetto)

Parametri	Tipo	Descrizione
Oggetto	Espr_	Campo
	o Espr_Var	Oggetto di destinazione

Contesto
Durante dell'inserimento delle procedure di formato, delle procedure di archivio e degli script.

Descrizione
VAI SU AREA introduce il punto d'inserzione nell'oggetto rappresentato da *Oggetto*. Questa procedura, ovviamente, ha senso soltanto nelle procedure di formato, nelle procedure di archivio o negli script di un formato in corso di inserimento. Se *Oggetto* non esiste nel formato in inserimento o se *Oggetto* non è inseribile, VAI SU AREA non esegue alcuna azione.

Esempio

Se(StatoCiv="Coniugata")	Se la persona è coniugata
VAI SU AREA(NomeRagazza)	Vai al campo NomeRagazza
altrimenti	altrimenti
VAI SU AREA(Funzione)	Vai al campo Funzione
Fine se	

Riferimenti
RIFIUTA

TESTO SELEZIONATO — GET HIGHLIGHT

Sintassi
TESTO SELEZIONATO(Oggetto;InizioSel;FineSel)

Parametri	Tipo	Descrizione
Oggetto	Espr_Var o Espr_Campo	Variabile o campo di cui volete ottenere la parte selezionata
InizioSel	Espr_Var	Posizione d'inizio selezione
FineSel	Espr_Var	Posizione di fine selezione

Contesto
Inserimento in un formato

Descrizione
TESTO SELEZIONATO viene usato per trovare il testo attualmente selezionato e restituisce la posizione del primo e dell'ultimo carattere selezionato. Potrete poi recuperare la stringa di caratteri corrispondente utilizzando la funzione Sotto_stringa.

I limiti restituiti sono: *InizioSel*=3; *FineSel*=6

Questo è un testo

I limiti restituiti sono: *InizioSel*=11; *FineSel*=11

TESTO SELEZIONATO restituisce il valore 1 in *InizioSel* e 2 in *FineSel* quando il campo selezionato è di tipo booleano o disegno.

Esempio

```
TESTO SELEZIONATO(Commento;vInizio;vFine)            Cerca un testo selezionato.
Se((vFine-vInizio)>0)                                Se la lunghezza del testo selezionato è superiore a 0.
    AVVISO(Sotto_stringa(Commento;vInizio;(vFine-vInizio)))   Visualizza la parte di testo selezionato
Fine se
```

Riferimenti
SELEZIONA TESTO

SELEZIONA TESTO

HIGHLIGHT TEXT

Sintassi
SELEZIONA TESTO(Oggetto;InizioSel;FineSel)

Parametri	**Tipo**	**Descrizione**
Oggetto	Espr_Var o Espr_Campo	Variabile o campo in cui viene impostata la selezione
InizioSel	Espr_Num	Posizione d'inizio selezione
FineSel	Espr_Num	Posizione di fine selezione

Contesto
Inserimento in un formato

Descrizione
SELEZIONA TESTO si posiziona sul campo o sulla variabile *Oggetto* e seleziona il testo compreso tra il carattere in posizione *InizioSel* fino al carattere in *FineSel*.
Se i 2 limiti, *InizioSel* o *FineSel*, si trovano oltre i limiti del testo o se *InizioSel* è maggiore di *FineSel*, SELEZIONA TESTO colloca il cursore a fine testo per i campi di tipo testo e seleziona tutto per gli altri tipi di campo.

Esempio

```
mPos:=Posizione("Buongiorno";IlTesto)
Se(mPos#0)
    SELEZIONA TESTO(IlTesto;mPos;mPos+Lunghezza("Buongiorno"))
Fine se
```

Cerchiamo la posizione di "Buongiorno" nel testo. Se "Buongiorno" compare almeno una volta, selezioniamo questa parola nel testo

Riferimenti
TESTO SELEZIONATO

Ultimo oggetto

Last area

Sintassi
Ultimo oggetto
➤ Puntatore

Contesto
Formato

Descrizione
Ultimo oggetto restituisce un puntatore all'ultimo oggetto (campo o variabile inseribile) modificato durante l'inserimento in un formato. Ultimo oggetto non punta alla zona in cui si trova il punto d'inserzione, ma all'ultima zona modificata. Tuttavia, se mettete il punto d'inserzione in un oggetto e fate clic, per esempio, su un pulsante, Ultimo oggetto restituirà un puntatore a questo campo. Le funzioni Tipo e Variabile vi permetteranno di definire con precisione l'oggetto puntato da Ultimo oggetto.

Esempio
Nel seguente formato d'inserimento, volete impostare un sistema di Aiuto per sapere cosa dovete inserire in ogni area. Usate il comando Ultimo oggetto per sapere in quale area vi trovate.

Lo script del pulsante Aiuto è il seguente:

```
Ultimo:=Ultimo oggetto
Caso fra
    :(Ultimo=(»[Clienti]Società))                          Ultimo = al puntatore verso Società?
    vMess:="Introduci il nome della Società")
    :(Ultimo=(»[Clienti]Indirizzo))                        Ultimo = al puntatore verso Indirizzo?
    vMess:="Introduci l'indirizzo del cliente, puoi andare a capo"
    :(Ultimo)=(»[Clienti]Sconto))                          Ultimo = al puntatore verso Sconto?
    vMess:="Introduci lo sconto concesso al cliente, tra 0 e 20%"
Fine caso
```

Riferimenti
Tipo, variabile

Tipo — Type

Sintassi
Tipo(Oggetto)
➤ Intero positivo

Parametri	Tipo	Descrizione
Oggetto	Espr_Campo o Espr_Var	Oggetto di cui si vuole conoscere il tipo

Descrizione
Tipo restituisce un valore compreso tra 0 e 23, corrispondente al tipo di *Oggetto*.

I valori indicati sono i seguenti:

Stringa	0	Vettore a due dimensioni	13
Reale	1	Vettore di reali	14
Testo	2	Vettore di interi	15
Disegno	3	Vettore di interi lunghi	16
Data	4	Vettore di date	17
Indefinito	5	Vettore di testi	18
Booleano	6	Vettore di immagini	19
Sotto_archivio	7	Vettore di puntatori	20
Intero	8	Vettore di stringhe	21
Intero Lungo	9	Vettore di booleani	22
Ora	11	Puntatore	23

Se volete conoscere il tipo di vettore, non potete scrivere **Tipo**(IlVettore), poichè in questo caso vi verrà indicato il vettore stesso, per esempio, se è un vettore di reali, 14. Se invece scrivete **Tipo**(IlVettore{1}), l'elemento del vettore verrà passato come parametro, e la funzione Tipo vi restituirà il valore che cercate, in questo caso, per un numero reale, 1.

Esempio
Questa funzione restituisce una stringa di caratteri contenente il testo selezionato nel campo corrente. Dato che è possibile selezionare una parte della zona di tipo numerico o data, esamineremo il tipo di oggetto corrente prima di indicare il valore selezionato dell'oggetto. Scriveremo una funzione chiamata *PrendiTesto*.

```
$0:=""
$Ultimo:=Ultimo oggetto                                   Ultimo è un puntatore all'oggetto corrente
TESTO SELEZIONATO($Ultimo»;$vInizio;$vFine)               Restituisci i limiti del testo selezionato
$IlTipo:=Tipo($Ultimo»)                                   IlTipo conterrà il tipo di oggetto corrente
Caso fra
    :($IlTipo=2) | ($IlTipo=0)`Stringa o testo
    $0:=Sotto_stringa($Ultimo»;$vInizio;($vFine-$vInizio))
    :($IlTipo=1) | ($IlTipo=4) | ($IlTipo=8) | ($IlTipo=9)`Numerico o data
    $0:=Sotto_stringa(Stringa($Ultimo»);$vInizio;($vFine-$vInizio))
    :($IlTipo=11)`Ora
    $0:=Sotto_stringa(Stringa in ora($Ultimo»);$vInizio;($vFine-$vInizio))
Fine caso
```

Riferimenti
Ultimo oggetto, Variabile, TESTO SELEZIONATO, SELEZIONA TESTO

Variabile   Is a variable

Sintassi
Variabile(Il Puntatore)
➤ Booleano

Parametri	Tipo	Descrizione
IlPuntatore	Puntatore	Puntatore a un oggetto

Descrizione
Variabile restituisce il valore Vero se l'oggetto puntato da *Puntatore* è una variabile.

Riferimenti
Tipo, Ultimo oggetto

RIDISEGNA   REDRAW

Sintassi
RIDISEGNA(Archivio | IlSottoarchivio)

Parametri	Tipo	Descrizione
Archivio	Espr_Archivio	Archivio incluso da ridisegnare
IlSottoarchivio	Espr_SottArch	Sottoarchivio da ridisegnare

Contesto
Inserimento in un formato

Descrizione
RIDISEGNA ridisegna sullo schermo l'oggetto di un formato.
Questo oggetto può essere un archivio incluso o un sotto-archivio.
In utilizzo corrente, non occorre chiamare questo comando. E' utile soltanto nei casi in cui 4th DIMENSION non può gestire in automatico la funzione di aggiornamento dello schermo, come ad esempio durante la modifica tramite procedura del campo di un archivio incluso.
Immaginate di far comparire un archivio [Righe] in un formato dell'archivio [Fatture]. Se modificate il valore di un campo dell'archivio incluso tramite programmazione, dovrete poi chiamare il comando RIDISEGNA per aggiornare la lista.

Modificato — Modified

Sintassi
Modificato(IlCampo)
➤ Booleano

Parametri	Tipo	Descrizione
IlCampo	Espr_Campo	Campo di cui volete verificare se è stato modificato

Contesto
Chiamata in Durante delle procedure di archivio o delle procedure di formato

Descrizione
Questa funzione restituisce il valore vero se *IlCampo* è stato modificato. Vi permette di verificare qual è il campo che ha causato la chiamata dei cicli d'inserimento. Non serve chiamare questa funzione all'interno dello script di un campo, poichè lo script viene attivato soltanto quando il campo è modificato.

Valore precedente — Old

Sintassi
Valore precedente(IlCampo)
➤ Valore in base al tipo di campo

Parametri	Tipo	Descrizione
IlCampo	Espr_Campo	Campo di cui si vuole conoscere il valore precedente

Descrizione
La funzione Valore precedente restituisce il valore precedente di *Campo*. Se inserite la scheda per la prima volta, Valore precedente restituisce un valore nullo, (0 per i numerici, "" per i testi, !00-00-00! per le date,...). Se state modificando schede già registrate, Valore precedente restituisce il valore di *Campo* al momento dell'ultima registrazione della scheda. Mentre inserite una scheda, 4th DIMENSION permette d'inserire in memoria una copia di questa scheda, e la funzione Valore precedente restituirà il valore del campo su disco.

Esempio
Questa funzione vi consente di stabilire se siete in fase di modifica o di inserimento delle schede. Immaginate un archivio [Clienti] con il campo NumCliente obbligatorio e sempre diverso da 0.

```
Se(Valore precedente(NumCliente)=0)      Si tratta di una nuova scheda
altrimenti                               altrimenti, la scheda é in modifica
Fine se
```

Questa funzione è anche utile nel caso di aggiornamenti eseguiti su relazioni. A questo proposito, fate riferimento alla sezione relativa ai comandi di gestione delle relazioni.

Riferimenti
RELAZIONE A UNO PRECEDENTE, RELAZIONE A MOLTI PRECEDENTE

IMPOSTA MASCHERA

SET FILTER

Sintassi
IMPOSTA MASCHERA(Zona;LaMaschera)

Parametri	Tipo	Descrizione
Zona	Espr_Var o Espr_Campo	Zona inseribile per la quale si vuole cambiare la maschera di inserimento
LaMaschera	Espr_Alfa	Maschera di sostituzione

Contesto
Inserimento in un formato

Descrizione
IMPOSTA MASCHERA sostituisce la maschera con *LaMaschera* per la zona d'inserimento *Zona* (campo o variabile) nel formato corrente. Se *Zona* è una stringa vuota, nessuna maschera sarà più associata a *Zona*. Fate riferimento al Manuale *Ambiente Progetto* per la descrizione della sintassi delle maschere. Questa modifica ha effetto soltanto sul formato a cui è stata applicata e non verrà quindi registrata. Se volete apportare una modifica definitiva alla maschera di *Zona*, modificate la maschera in ambiente Progetto. Se chiamate questo comando per la zona in corso di inserimento, la modifica diventerà effettiva soltanto quando uscirete dalla zona.

Esempio
Durante l'inserimento di un campo data, volete utilizzare una maschera che permetta di inserire soltanto il giorno nella forma seguente: "Giorno / Numero del mese / Numero dell'anno". Inoltre, volete che questa maschera sia utilizzabile in un formato europeo (Giorno/Mese/Anno) o americano (Mese/Giorno/Anno).

```
Se(Prima)
    Caso fra
    :(Nazione="Italia")
    LaMaschera:="&9  ##"+Sotto_stringa(Stringa([Clienti]Data);3;6)
    :(Nazione="USA")
    LaMaschera:="&9  "+Stringa(Mese di([Cliente]Data))+"/##/"+Stringa(Anno di ([Clienti]Data))
    Fine caso
    IMPOSTA MASCHERA([Clienti]Data;LaMaschera)
Fine se
```

Riferimenti
IMPOSTA FORMATTAZIONE

IMPOSTA LISTA SCELTE — SET CHOICE LIST

Sintassi
IMPOSTA LISTA SCELTE(Zona;ListaScelte)

Parametri	Tipo	Descrizione
Zona	Espr_Var o Espr_Campo	Zona a cui associare una lista scelte
ListaScelte	Espr_Alfa	Nome di una lista scelte

Contesto
Inserimento in un formato

Descrizione
IMPOSTA LISTA SCELTE associa la lista scelte *ListaScelte* alla zona d'inserimento *Zona* (campo o variabile), che compare in un formato sullo schermo. Se questa lista non esiste, IMPOSTA LISTA SCELTE non esegue alcuna azione.
Questa modifica ha effetto soltanto sul formato a cui è applicata, quindi non verrà registrata. Se volete apportare una modifica definitiva alla lista scelte di *Zona*, modificatela quando siete nell'ambiente **Progetto**. Se chiamate questo comando per la zona in corso di inserimento, la modifica diventerà effettiva soltanto quando uscirete dalla zona.

Esempio
Scriviamo una procedura di formato che modifichi la lista scelte collegata al campo Posta in base alla nazione scelta.

```
Se(Nazione="Italia")
    IMPOSTA LISTA SCELTE(Posta;"Posta_IT")
Fine se
```

Riferimentil
IMPOSTA MASCHERA, IMPOSTA SE INSERIBILE

IMPOSTA SE INSERIBILE — SET ENTERABLE

Sintassi
IMPOSTA SE INSERIBILE(Zona;Inseribile)

Parametri	Tipo	Descrizione
Zona	Espr_Var o Espr_Campo	Zona da impostare
Inseribile	Espr_Bool	Vero per inseribile, Falso per non inseribile

Contesto
Inserimento in un formato. Nel caso di un campo, questo dev'essere definito come non inseribile in ambiente Progetto.

Descrizione
IMPOSTA SE INSERIBILE imposta il valore dell'attributo inseribile per una zona d'inserimento *Zona* che compare nel formato visualizzato. Se *Inseribile* è vero, la zona sarà inseribile, altrimenti sarà non inseribile. Questa modifica ha effetto soltanto sul formato visualizzato sullo schermo e non viene registrata. Se chiamate questo comando per la zona in corso di inserimento, la modifica diventerà effettiva solo quando uscirete dalla zona.

Esempio
Volete disabilitare l'inserimento nel seguente formato finché non verrà inserito il campo CodCliente.

Ecco lo script del campo CodCliente:

```
Se (CodCliente#"")                                  Se il campo CodCliente è stato inserito
    IMPOSTA SE INSERIBILE(Società;Vero)             Autorizza l'inserimento
    IMPOSTA SE INSERIBILE(Indirizzo;Vero)
    IMPOSTA SE INSERIBILE(Sconto;Vero)
altrimenti                                          Il campo CodCliente è vuoto, disabilita l'in-
    IMPOSTA SE INSERIBILE(Società;Falso)            serimento.
    IMPOSTA SE INSERIBILE(Indirizzo;Falso)
    IMPOSTA SE INSERIBILE(Sconto;Falso)
Fine se
```

Riferimenti
IMPOSTA FORMATTAZIONE

IMPOSTA FORMATTAZIONE — SET FORMAT

Sintassi
IMPOSTA FORMATTAZIONE(Zona;Formattazione)

Parametri	Tipo	Descrizione
Zona	Espr_Var o Espr_Campo	Area di inserimento di cui volete cambiare la formattazione
Formattazione	Espr_Alfa	Formattazione con cui viene visualizzata l'area

Contesto
Inserimento in un formato

Descrizione
IMPOSTA FORMATTAZIONE modifica la formattazione dell'area di inserimento *Zona* nel formato corrente, sostituendola con la nuova formattazione *Formattazione*. Questa modifica ha effetto soltanto sul formato corrente e non verrà registrata. Se avete chiamato questo comando per l'area in corso di inserimento, la modifica diventerà effettiva soltanto quando uscirete dall'area.

Esempio

```
Se(Nazione="USA")
   IMPOSTA FORMATTAZIONE([Fattura]Importo;"US $ ### ### ##0,00")
Fine se
```

Casi particolari: aree di tipo data, ora e disegno

Date: 4 formattazioni		Ore: 5 formattazioni		Disegni: 3 formattazioni	
Corto	(1)	o:mn:s	(1)	Rifilato	(1)
Abbreviato	(2)	o:mn	(2)	Adattato	(2)
Lungo	(3)	Ore:Minuti:Secondi	(3)	Sullo sfondo	(3)
Speciale	(4)	Ore:Minuti	(4)		
		o:mn AM/PM	(5)		

Se Zona è un campo:
Formattazione dovrà essere un carattere il cui codice ASCII sarà uguale al numero della formattazione desiderata.
Per esempio, Carattere(2) indicherà un disegno con formattazione di tipo Adattato.

Se Zona è una variabile:
Date: carattere il cui codice ASCII è uguale alla formattazione richiesta
Ore: carattere il cui codice ASCII è uguale alla formattazione richiesta +8
Disegni: carattere il cui codice ASCII è uguale alla formattazione richiesta +16
Per esempio, Carattere (2+8) sarà il parametro di formattazione per una variabile di tipo ora da formattare in o:mn.

IMPOSTA COLORE — SET COLOR

Sintassi
IMPOSTA COLORE(Oggetto;Colore)

Parametri	Tipo	Descrizione
Zona	Espr_Var o Espr_Campo	Zona di inserimento in cui verrà impostato il colore
Colore	Espr_Num	Colore visualizzato

Contesto
Inserimento in un formato, stampa

Descrizione
IMPOSTA COLORE modifica il colore di un oggetto inserito in un formato.

IMPOSTA COLORE ha effetto sulle variabili, sui campi, sui pulsanti standard, sui radio bottoni e sui pulsanti d'opzione, sulle aree di scorrimento e sui menu a comparsa.

Per un Macintosh a 256 colori, il colore viene espresso nel modo seguente:
Colore=-(PrimoPiano+(256*Sfondo))
PrimoPiano e Sfondo sono valori compresi tra 1 e 256 e corrispondono al numero del colore nella tavolozza dei colori corrente.

Per un Macintosh ad 8 colori, il colore viene espresso nella forma seguente:
Colore=PrimoPiano+(8*Sfondo)

Gli attributi dei colori sono temporanei, e non vengono registrati. Se desiderate impostare il colore di un oggetto in modo definitivo, servitevi della finestra di formato nell'ambiente Progetto.

Esempio
In una fatturazione, facciamo variare il colore della zona Scadenza in base ai termini di pagamento

```
Caso fra
    :(Termine<30)                          Il termine è inferiore a 30 giorni
    IMPOSTA COLORE(Scadenza;-9)            Imposta il colore verde
    (Termine<60)                           Il termine è inferiore a 60 giorni
    IMPOSTA COLORE(Scadenza;-7)            Imposta il colore blu
altrimenti                                 Altrimenti il termine sarà soltanto superiore a 60 giorni
    IMPOSTA COLORE(Scadenza;-4)            Imposta il rosso
Fine caso
```

Riferimenti
IMPOSTA STILE, IMPOSTA DIMENSIONE, IMPOSTA FONT

CAMBIA TESTO PULSANTE — BUTTON TEXT

Sintassi
CAMBIA TESTO PULSANTE(Pulsante;Testo)

Parametri	Tipo	Descrizione
Pulsante	Espr_Var	Nome della variabile associata al pulsante
Testo	Espr_Alfa	Nuovo testo del pulsante

Contesto
Inserimento in un formato, visualizzazione su schermo

Descrizione
CAMBIA TESTO PULSANTE modifica il testo del pulsante e lo sostituisce con *Testo*. Il pulsante verrà automaticamente aggiornato con il nuovo testo.
CAMBIA TESTO PULSANTE ha effetto soltanto sul formato corrente, e non modifica il testo del pulsante nella struttura della base dati. Se in seguito il pulsante verrà richiamato, verrà visualizzato con il testo originale, indicato al momento della creazione del formato.

Esempio
In questo esempio viene cambiato il testo di un pulsante in base alla lingua da usare

Caso fra	
:(Lingua="Francese")	Se la lingua è il francese
CAMBIA TESTO PULSANTE(pAnnulla;"Annuler")	Il pulsante Annulla si chiamerà "Annuler"
:(Lingua="Inglese")	Se la lingua è l'inglese
CAMBIA TESTO PULSANTE(pAnnulla;"Cancel")	Il pulsante Annulla si chiamerà "Cancel"
Fine caso	

Riferimenti
ABILITA PULSANTE, DISABILITA PULSANTE

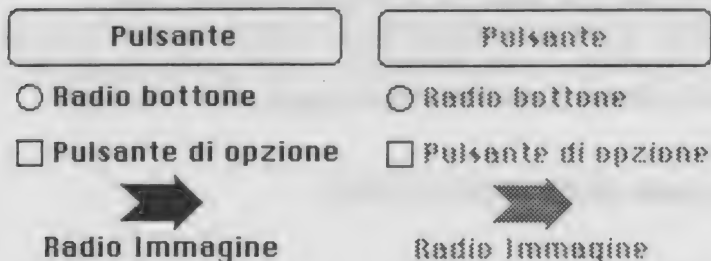
DISABILITA PULSANTE — DISABLE BUTTON

Sintassi
DISABILITA PULSANTE(Pulsante)

Parametri	**Tipo**	**Descrizione**
Pulsante	Espr_Alfa	Pulsante da disabilitare

Contesto
Inserimento in un formato, visualizzazione

Descrizione
DISABILITA PULSANTE disattiva il pulsante *Pulsante*. Il pulsante verrà aggiornato automaticamente e apparirà a luminosità ridotta; facendo clic su questo pulsante non verrà eseguita alcuna azione.
DISABILITA PULSANTE può essere applicato a tutti i tipi di pulsante: pulsanti standard, pulsanti invisibili, pulsanti ad inversione, radio bottoni, pulsanti d'opzione e radio immagini.

Pulsante	Pulsante
○ Radio bottone	○ Radio bottone
□ Pulsante di opzione	□ Pulsante di opzione
Radio Immagine	Radio Immagine

DISABILITA PULSANTE ha effetto soltanto sul formato corrente. Se in seguito il formato verrà richiamato, il pulsante sarà abilitato.
Potete ovviamente chiamare DISABILITA PULSANTE nel ciclo di Prima di un formato. In questo modo, il pulsante sarà disabilitato quando il formato verrà visualizzato.

Nota
E' sempre preferibile disabilitare un pulsante piuttosto di permettere un'operazione non corretta che darà luogo ad un messaggio di errore sullo schermo.

Esempio
Questo breve esempio vi descrive come utilizzare i comandi ABILITA PULSANTE e DISABILITA PULSANTE per controllare l'inserimento.
Se CodCliente è stato inserito, il pulsante di conferma *Conferma* si abiliterà, altrimenti resterà disabilitato e non sarà possibile confermare la scheda.

```
Se(CodCliente#"")
    ABILITA PULSANTE(Conferma)
altrimenti
    DISABILITA PULSANTE(Conferma)
Fine se
```

Riferimenti
ABILITA PULSANTE, CAMBIA TESTO PULSANTE .

ABILITA PULSANTE — ENABLE BUTTON

Sintassi
ABILITA PULSANTE(Pulsante)

Parametri	Tipo	Descrizione
Pulsante	Espr_Var	Pulsante da abilitare

Contesto
Inserimento in un formato, visualizzazione

Descrizione
ABILITA PULSANTE attiva il pulsante *Pulsante*. Il pulsante viene aggiornato automaticamente e apparirà a luminosità normale, e facendo clic su questo pulsante verrà eseguita l'azione corrispondente.
ABILITA PULSANTE può essere applicato a tutti i tipi di pulsante: pulsanti standard, pulsanti invisibili, pulsanti ad inversione, radio bottoni, pulsanti d'opzione e radio immagini. In condizioni normali, tutti i pulsanti di un formato vengono automaticamente abilitati quando il formato appare sullo schermo.

Esempio
Fate riferimento all'esempio che descrive il comando DISABILITA PULSANTE

Riferimenti
DISABILITA PULSANTE, CAMBIA TESTO PULSANTE

IMPOSTA FONT FONT

Sintassi
IMPOSTA FONT(Oggetto;TipoCar)

Parametri	Tipo	Descrizione
Oggetto	Espr_Var o Espr_Campo	Oggetto di cui si vuole cambiare il tipo di carattere
TipoCar	Espr_Num o Espr_Alfa	Nome o numero del tipo di carattere

Contesto
Stampa, inserimento in un formato

Descrizione
IMPOSTA FONT modifica il tipo di carattere utilizzato per la visualizzazione o la stampa di una variabile o di un campo specificato da *Oggetto*. *TipoCar* può contenere sia il numero sia il nome del tipo di carattere che volete utilizzare.

Nome del tipo	Numero del font
Chicago (tipo di carattere di sistema)	0
Geneva (tipo di carattere predeterminato di 4th DIMENSION)	1
New York	2
Geneva	3
Monaco	4
Venice	5
London	6
Athens	7
San Francisco	8
Toronto	9
Cairo	11
Los Angeles	12
Times	20
Helvetica	21
Courier	22
Symbol	23
Taliesin	24

Se indicate un tipo di carattere non disponibile, *Oggetto* viene visualizzato con il tipo di carattere specificato al momento della creazione del formato.
IMPOSTA FONT ha effetto solo sul formato corrente. Se in seguito il formato verrà richiamato, *Oggetto* verrà visualizzato o stampato con il tipo di carattere specificato al momento della creazione del formato.

Esempio
IMPOSTA FONT(vNome;21) Camba vNome in Helvetica

Riferimenti
IMPOSTA DIMENSIONE, IMPOSTA STILE

IMPOSTA DIMENSIONE — FONT SIZE

Sintassi
IMPOSTA DIMENSIONE(Oggetto;NuovaDim)

Parametri	Tipo	Descrizione
Oggetto	Espr_Var o Espr_Campo	Oggetto di cui si vuole cambiare il passo di scrittura
NuovaDim	Espr_NumPos	Passo di scrittura in punti

Contesto
Stampa

Descrizione
IMPOSTA DIMENSIONE modifica il passo di scrittura di *Oggetto* sostituendovi quello stabilito da *NuovaDim*. Questo può variare da 1 a 128 punti. Se indicate un passo di scrittura negativo, IMPOSTA DIMENSIONE non esegue alcuna azione.
Se invece indicate un passo di scrittura che supera il limite massimo, IMPOSTA DIMENSIONE visualizzerà *Oggetto* con il passo di scrittura massimo di 128 punti.
Se non specificate il passo di scrittura, IMPOSTA DIMENSIONE utilizza il passo di scrittura corrente di *Oggetto* indicato al momento della creazione del formato.
IMPOSTA DIMENSIONE agisce solo sul formato corrente. Se in seguito richiamerete il formato, *Oggetto* verrà visualizzato o stampato con il passo di scrittura indicato al momento della creazione del formato.

Esempio

IMPOSTA DIMENSIONE(vNom;12) — Cambia il passo di scrittura di vNome

Riferimenti
IMPOSTA FONT, IMPOSTA STILE

IMPOSTA STILE — FONT STYLE

Sintassi
IMPOSTA STILE(Oggetto;NuovoStile)

Parametri	Tipo	Descrizione
Oggetto	Espr_Var o Espr_Campo	Oggetto di cui si vuole modificare lo stile
NuovoStile	Espr_NumPos	Stile dell'oggetto

Contesto
Inserimento, visualizzazione, stampa

Descrizione
IMPOSTA STILE modifica lo stile di visualizzazione o di stampa di *Oggetto*, sostituendovi lo stile specificato da *NuovoStile*. Il valore di *NuovoStile* è calcolato sommando i valori dei coefficienti elencati qui di seguito:

Stile	**Coefficiente**
Testo Normale	0
Grassetto	1
Corsivo	2
Sottolineato	4
Bordato	8
Ombreggiato	16
Compresso	32
Espanso	64

Esempio
Volete che una variabile appaia in corsivo, sottolineato, espanso. Scrivete:

IMPOSTA STILE(Var;(2+4+64))

Avreste potuto scrivere anche IMPOSTA STILE(Var;70), ma la lettura di questa presentazione risulterebbe più difficile.

Un consiglio:
Se non volete utilizzare questi codici numerici, potete creare delle variabili globali nella procedura Startup, nel modo seguente:

vTestoNormale:=0
vGrassetto:=1
vCorsivo:=2
vSottolineato:=4
vBordato:=8
vOmbreggiato:=16
vCompresso:=32
vEspanso:=64

Successivamente, potete scrivere:
IMPOSTA STILE(Var;(vGrassetto+vSottolineato+vEspanso))

Esempio
Prendiamo un archivio dotato del seguente formato di tipo Lista:

Designazione	Riferimento	Prezzo	Vendite	Risultato
Designazione	Riferimento	Prezzo	Vendite	v

Vogliamo, per ogni scheda ed in base alle vendite, visualizzare un commento nella colonna risultato, in uno stile diverso a seconda del tipo di commento.

Associamo quindi alla variabile v il seguente script:

```
Caso fra
    :(Vendite<1000)
    Stile:=vCorsivo
    v:="Scarso"
    :(Vendite>=1000)&(Vendite<5000)
    Stile:=vGrassetto+vCorsivo
    v:="Medio"
    :(Vendite>=5000)
    Stile:=vBordato
    v:="Ottimo"
Fine caso
CAMBIA STILE(v;Stile)
```

La visualizzazione o la stampa di una selezione assume l'aspetto seguente:

Designazione	Riferimento	Prezzo	Vendite	Risultato
Macchinine pompieri	MP7002	25500 Lit	5200 p	**Ottimo**
Portachiavi Topolino	PT1489	1000 Lit	4200 p	***Medio***
Figurine	FG5286	500 Lit	700 p	*Scarso*
Palloni rugby	RG4194	15000 Lit	5420 p	**Ottimo**

Riferimenti
IMPOSTA FONT, IMPOSTA DIMENSIONE

INVERTI SFONDO — INVERT BACKGROUND

Sintassi
INVERTI SFONDO(Oggetto)

Parametri	Tipo	Descrizione
Oggetto	Espr_Var	Variabile di cui si vuole invertire lo sfondo

Contesto
Stampa di un formato (tranne che nel caso di stampanti PostScript), inserimento e visualizzazione delle schede.

Descrizione
INVERTI SFONDO inverte lo sfondo della variabile *Oggetto*, quindi visualizza *Oggetto* in bianco su sfondo nero. Attenzione: non potete stampare una variabile a cui è stata applicata l'inversione su una stampante PostScript: il driver di stampa Macintosh di sistema non può gestire questa funzione.

Esempio

INVERTI SFONDO(LaVar)

Riferimenti
IMPOSTA COLORE

STAMPA SELEZIONE

PRINT SELECTION

Sintassi
STAMPA SELEZIONE({Archivio};{*})

Parametri	Tipo	Descrizione
Archivio	Espr_Archivio	Archivio contenente la selezione da stampare
*	Costante	Richiesta di non visualizzazione delle finestre di dialogo per la stampa

Descrizione
STAMPA SELEZIONE avvia la stampa delle schede della selezione corrente di *Archivio* utilizzando il formato Lista corrente. *Archivio* è opzionale. Se questo argomento viene omesso, STAMPA SELEZIONE si riferirà all'archivio corrente.
Se si aggiunge l'argomento * al comando STAMPA SELEZIONE, le due finestre di dialogo standard per la stampa non verranno visualizzate. La stampa viene eseguita usando il formato di stampa assegnato al formato al momento della sua creazione, oppure usando quello del formato impostato dal comando IMPOSTA PARAMETRI DI STAMPA. Il tipo di uscita (su stampante o su schermo) sarà quello dell'ultima stampa effettuata.
Potete sapere se la stampa si è conclusa senza interruzioni verificando la variabile OK: OK assumerà il valore 0 se avete fatto clic sul pulsante **Annulla** di una delle due finestre di dialogo per la scelta dei parametri di stampa, oppure se avete fatto clic su **Interrompi** durante la stampa. Se la stampa è terminata correttamente, OK assumerà il valore 1.

Potete controllare se 4th DIMENSION sta stampando la prima intestazione servendovi della funzione Inizio selezione o se sta stampando l'ultimo pié di pagina per mezzo della funzione Fine selezione. Fate riferimento anche all'appendice della prima parte di questo Manuale relativa ai cicli di esecuzione.
Per sapere come stampare un rapporto che presenta dei sotto totali, fate riferimento ai comandi LIVELLI DI ROTTURA e ACCUMULA SU descritti successivamente. Troverete un esempio dettagliato alla descrizione della funzione Sotto totale.

Esempio

STAMPA SELEZIONE	Stampa la selezione corrente dell'archivio corrente
STAMPA SELEZIONE([Clienti];*)	Stampa senza aprire le finestre di dialogo per la stampa

Riferimenti
STAMPA FORMATO, STAMPA ETICHETTE, LIVELLI DI ROTTURA, ACCUMULA SU

STAMPA ETICHETTE — PRINT LABEL

Sintassi
STAMPA ETICHETTE({Archivio};{*})
STAMPA ETICHETTE({Archivio};NomeEtich;{*})

Parametri	Tipo	Descrizione
Archivio	Espr_Archivio	Archivio contenente la selezione da stampare
*	Costante	Richiesta di non visualizzazione delle finestre di dialogo per la stampa
NomeEtich	Espr_Alfa	Nome del documento etichette

Descrizione
STAMPA ETICHETTE stampa le schede della selezione corrente dell'archivio *Archivio* sotto forma di etichette.

1. Nella prima sintassi, STAMPA ETICHETTE utilizza il formato Lista in uso dell'archivio *Archivio*. Il numero di etichette stampate una accanto all'altra dipende dalla posizione del margine destro del righello orizzontale del formato, che indica la larghezza di un'etichetta in funzione del formato di stampa scelto.
4th DIMENSION calcola il numero di etichette stampabili in larghezza. In fase di stampa viene considerato soltanto il dettaglio del formato: quindi vi consigliamo di collocare il margine dell'intestazione del formato a 0 e quelli di rottura e di pié di pagina al livello del dettaglio del formato.
Archivio è un argomento opzionale. Se viene omesso, STAMPA ETICHETTE si riferirà all'archivio corrente. L'argomento opzionale * ha la stessa funzione descritta nel comando STAMPA SELEZIONE.

2. Nella seconda sintassi, STAMPA ETICHETTE utilizza il formato di etichette stabilito dalla finestra di dialogo per la creazione delle etichette disponibile in ambiente **Utilizzo** per stampare le schede della selezione corrente dell'archivio *Archivio*.

- Se *NomeEtich* non è una stringa vuota e nessun documento ha questo nome, 4th DIMENSION vi propone la finestra di dialogo per la creazione delle etichette. Se volete assicurarvi che la finestra di dialogo per la creazione delle etichette venga aperta, usate, ad esempio, la stringa Carattere(1). Infatti, questo carattere è inaccessibile da tastiera e non è quindi possibile che esista alcun documento con questo nome.

- Se *NomeEtich* non è una stringa vuota e corrisponde al nome di un documento esistente, STAMPA ETICHETTE carica in memoria la descrizione delle etichette e le stampa utilizzando i parametri di stampa scelti al momento della definizione delle etichette.

- Se *NomeEtich* è una stringa vuota, STAMPA ETICHETTE visualizzerà la finestra di dialogo standard per l'apertura dei documenti, con cui potrete scegliere il documento etichette desiderato.

Esempio

IMPOSTA ARCHIVIO CORRENTE([Clienti])	Stiamo lavorando sull'archivio Clienti
TUTTE LE SCHEDE	Seleziona tutte le schede
ORDINA([Clienti]CAP;>)	Ordina i clienti in base ai loro codici postali
DIALOGO("StampaEtich")	Quale tipo di stampa desideri realizzare?
Caso fra	
:(pNuova=1)	Scegli un documento etichette già creato
STAMPA ETICHETTE("")	
:(pStandard=1)	Apri un formato etichette già pronto
STAMPA ETICHETTE("Standard")	
(pMia=1)	Crea nuove etichette
STAMPA ETICHETTE(Carattere(1))	
Fine caso	

Riferimenti
STAMPA SELEZIONE, STAMPA FORMATO

Pagina in stampa — Printing page

Sintassi
Pagina in stampa
➤ Numerico

Descrizione
Pagina in stampa fornisce il numero della pagina in stampa. Questa funzione consente la numerazione automatica delle pagine di una stampa in corso.

Esempio
Script della variabile vNr che si trova nel formato da stampare

vNr:=**Pagina in stampa** — vNr assume il valore del numero della pagina in stampa

Riferimenti
STAMPA SELEZIONE

LIVELLI DI ROTTURA — BREAK LEVEL

Sintassi
LIVELLI DI ROTTURA(NrLiv;{CambioPagina})

Parametri	Tipo	Descrizione
NrLiv	Espr_NumPos	Numero dei livelli di rottura
CambioPagina	Espr_NumPos	Livello di rottura che provoca il cambio pagina

Descrizione
Il comando LIVELLI DI ROTTURA deve essere chiamato prima di una chiamata del comando STAMPA SELEZIONE. Vi permette di determinare il numero dei livelli di rottura della vostra stampa.
Questo numero deve essere inferiore o uguale ai livelli di ordinamento effettuati prima della stampa.
L'argomento opzionale *CambioPagina* provoca un cambio pagina sul livello di rottura scelto.
4th DIMENSION non è predisposto per inserire cambi pagina anticipati.

Esempio
Stampa di un rapporto di fatture con rotture inserite in base al cliente ed al mese.

```
IMPOSTA ARCHIVIO CORRENTE([Fatture])
TUTTE LE SCHEDE
ORDINA([Fatture]Clienti;>;[Fatture]Mese;>)     Ordina le fatture in base al cliente ed al mese
LIVELLI DI ROTTURA(2;1)                        Inserisci due livelli di rottura e un cambio pagina per ogni Cliente
STAMPA SELEZIONE                               Stampa le fatture
```

Riferimenti
ACCUMULA SU, STAMPA SELEZIONE

ACCUMULA SU — ACCUMULATE

Sintassi
ACCUMULA SU(Oggetto1;...;OggettoN)

Parametri	Tipo	Descrizione
Oggetto1...N	Espr_Campo o Espr_Var	Campo o variabile da accumulare

Descrizione
ACCUMULA SU è un comando che deve essere chiamato prima di stampare una selezione, se volete effettuare dei calcoli, ad esempio dei sotto totali. Fate riferimento all'esempio illustrato nella descrizione della funzione Sotto totale.

Esempio
Stampa di un rapporto di fatture con rotture in base al cliente ed al mese.

```
IMPOSTA ARCHIVIO CORRENTE([Fatture])
TUTTE LE SCHEDE
FORMATO LISTA("RapportoFatture")               Seleziona il formato per la stampa
ORDINA([Fatture]Clienti;>;[Fatture]Mese;>)     Ordina le fatture per cliente e per mese
LIVELLI DI ROTTURA(2;1)                        Inserisci due livelli di rottura e un cambio pagina per ogni Cliente
ACCUMULA SU([Fattura]TotaleNetto)              Effettua dei calcoli su [Fatture]TotaleNetto
STAMPA SELEZIONE                               Stampa le fatture
```

Il vostro formato dovrà contenere 1 variabile vTot che comparirà nelle tre zone R0, R1 ed R2.

Se(**In Rottura**)	Se stampi una rottura
vTot:=**Sotto totale**([Fatture]TotaleNetto)	Calcola vTot
Fine se	

Riferimenti
LIVELLI DI ROTTURA, STAMPA SELEZIONE

IMPOSTA PARAMETRI DI STAMPA — PAGE SETUP

Sintassi
IMPOSTA PARAMETRI DI STAMPA({Archivio};NomeFormato)

Parametri	Tipo	Descrizione
Archivio	Espr_Archivio	Archivio contenente il formato
NomeFormato	Espr_Alfa	Nome del formato

Descrizione
IMPOSTA PARAMETRI DI STAMPA deve essere chiamato prima dei comandi STAMPA SELEZIONE o STAMPA ETICHETTE e permette di specificare se la stampa deve essere effettuata usando i parametri stabiliti per il formato *NomeFormato*. Questo comando viene attivato soltanto al primo comando di stampa che incontrate.

Archivio è un argomento opzionale. Se viene omesso, IMPOSTA PARAMETRI DI STAMPA si riferirà all'archivio corrente.

Esempio

IMPOSTA ARCHIVIO CORRENTE([Clienti])	
TUTTE LE SCHEDE	
FORMATO LISTA("ListaClienti")	Seleziona il formato di stampa
IMPOSTA PARAMETRI DI STAMPA("Italia")	Utilizza i parametri di stampa di "Italia"
STAMPA SELEZIONE(*)	Stampa senza aprire le finestre di dialogo per la stampa

Riferimenti
STAMPA SELEZIONE, STAMPA ETICHETTE

Livello — Level

Sintassi
Livello
➤ Intero

Contesto
Ciclo: In rottura durante una stampa

Descrizione
Livello restituisce il livello di rottura corrente durante una stampa. Se Livello restituisce il valore 0, significa che 4th DIMENSION sta stampando il totale generale. Se Livello restituisce il valore 1, 4th DIMENSION sta stampando il sotto totale del livello 1 (ossia una rottura sul primo criterio di ordinamento), e così via.

Esempio

Dopo un ordinamento di [Rapporti]Nazione, [Rapporti]Regione e [Rapporti]PR, stampiamo un rapporto e vogliamo ottenere dei sotto totali.

```
Se(In rottura)
   Caso fra
      :(Livello =0)
      vTit:="Totale Generale"
      :(Livello=1)
      vTit:="Sotto totale per"+[Rapporti]Nazione
      :(Livello=2)
      vTit:="Sotto totale per"+[Rapporti]Regione
   Fine caso
Fine se
```

Riferimenti
Sotto totale, STAMPA SELEZIONE

Sotto totale

Subtotal

➤ Numerico

Sintassi

Sotto totale(IlCampo;{CambioPagina})

Parametri	Tipo	Descrizione
IlCampo	Espr_Campo	Campo su cui calcolare il sotto totale
CambioPagina	Espr_Num	Livello di rottura per i cambi pagina

Contesto

Ciclo di In rottura di un comando STAMPA SELEZIONE

Descrizione

Sotto totale effettua il sotto totale del campo *IlCampo* durante la stampa di una selezione di schede ordinate. Dovete chiamare Sotto totale nel ciclo In rottura delle procedure di formato del formato in stampa. *IlCampo* deve essere di tipo numerico, intero o intero lungo. Per visualizzare il sotto totale, dovete creare una variabile che inserirete nella zona di rottura del formato di stampa.

CambioPagina è opzionale e vi permette di stabilire a quale livello di rottura volete effettuare un cambio pagina. Se questo argomento viene omesso, la stampa avrà luogo senza cambio pagina.
Non dimenticate che per ottenere dei sotto totali a n livelli, dovrete ordinare su n+1 livelli. Diversamente dalla funzione Somma, Sotto totale viene applicato soltanto in fase di stampa.

Esempio

```
Se(In Rottura)
   v:=Sotto totale([Clienti]GA fatturato)
   Caso fra
      :(Livello=0)
      vTit:="Totale Generale"
      :(Livello=1)
      vTit:="Sotto totale per la regione"+[Clienti]Regione
   Fine caso
Fine se
```

Esempio dettagliato
Prendiamo in esame la seguente base dati:

Elenco schede

Pittore	Titolo	Anno	Qtà	Valore
Kandinsky	Punte nell'arco	1912	7	3668000
Brueghel	Danza di contadini	1568	3	4500000
Kandinsky	Acquarello astratto	1910	3	2355000
Brueghel	Giochi di fanciulli	1560	5	3910000
Brueghel	Raccolta del grano	1565	8	6256000
Kandinsky	Composizione n.7	1913	2	2400000
Van Gogh	Cipressi	1890	9	8595000
Van Gogh	Caffé di notte	1888	2	1120000
Van Gogh	La chiesa di Auvers	1890	5	2800000

Struttura

Poster	
Pittore	A
Titolo	A
Quantità	I
Valore	R
Commenti	T
Anno	I

Una scheda in formato Pagina

Riproduzioni	
Pittore	Brueghel
Titolo	Danza di contadini
Quantità	3
Valore	4500000
Anno	1568
Commenti	Detto il Vecchio. Pittore fiammingo (1525-1569). Scene di strada.

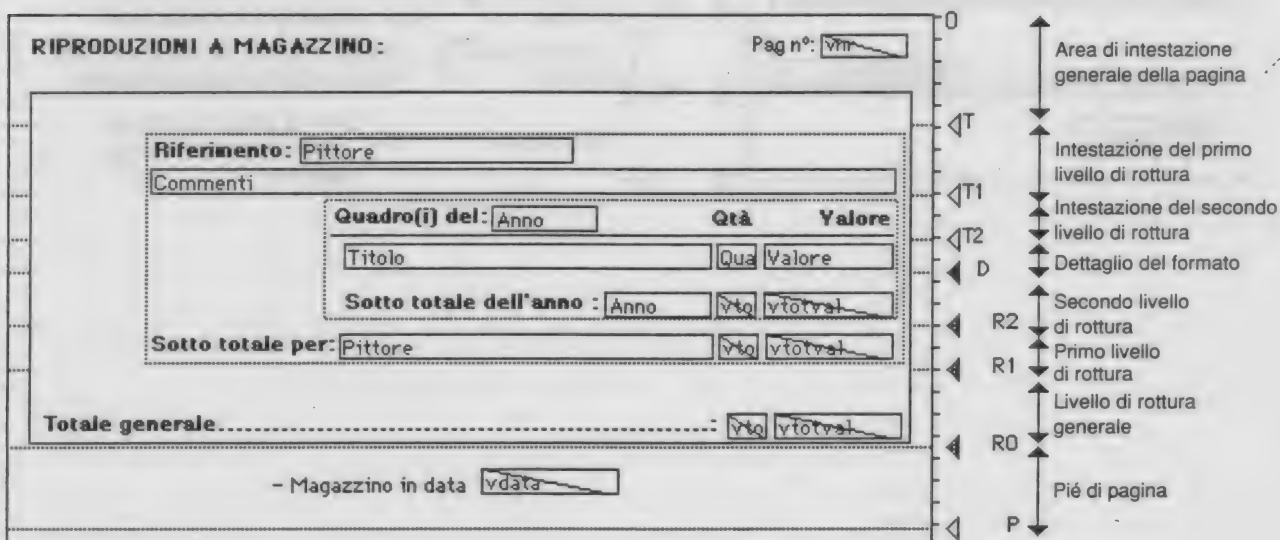
Vogliamo avere una visione generale ed ordinata del nostro magazzino di riproduzioni, con due livelli di rottura e intestazioni specifiche.
Il dettaglio del formato è costituito dal campo Quantità e dal campo Prezzo unitario.
Poichè la particolarità di un rapporto è quella di poter calcolare i valori del magazzino, facciamo un calcolo a livello del campo Valore, e precisamente:
Valore:=Prezzo unitario*Quantità.
Dato che il calcolo viene eseguito nello script del campo, che viene sempre chiamato in quanto si trova nel dettaglio del formato, non è più necessario calcolare questo valore negli script delle variabili delle zone di rottura.
Per ottenere l'esecuzione di questo script a tutti i livelli, dovete disattivare il pulsante d'opzione **Esegui solo se "Durante".**

Come organizzare il vostro formato?
Partiamo dal dettaglio del formato e risaliamo per creare le intestazioni, proseguiamo verso il basso per visualizzare le rotture.
Potete visualizzare un'intestazione specifica per ogni livello di rottura.

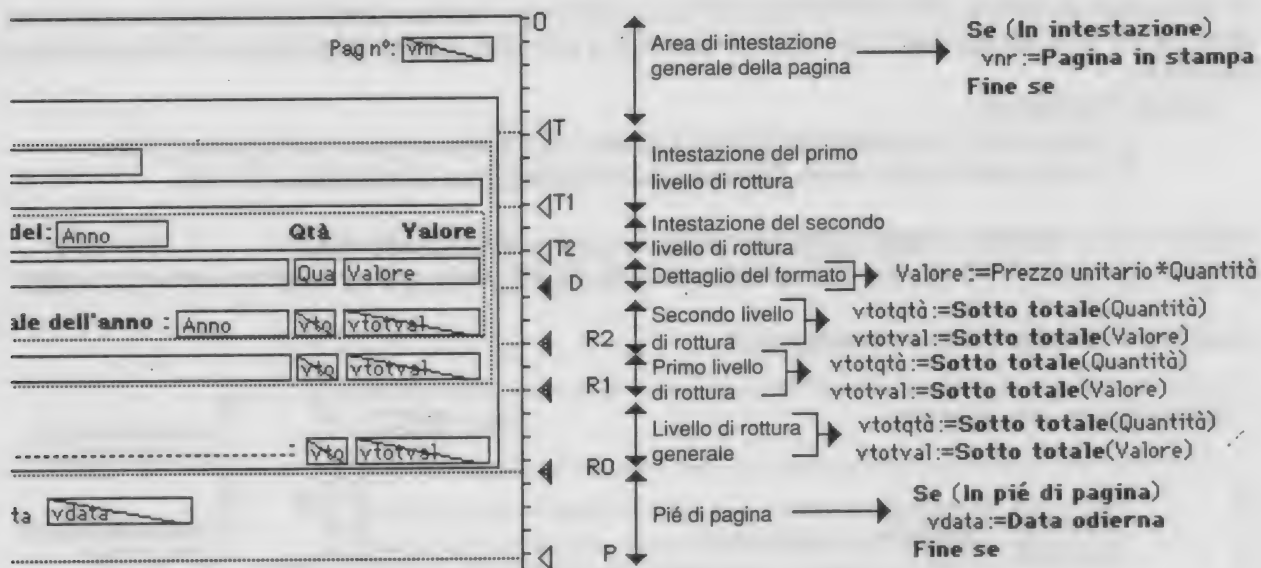
Costruiamo ora le rotture con intestazioni associate.
Vogliamo:
- Un'intestazione generale della pagina
  Questa intestazione ha un titolo, RIPRODUZIONI A MAGAZZINO e la numerazione delle pagine.
  La variabile vnr viene calcolata da uno script
- Un pié di pagina, che presenta la data del giorno ed il titolo "Magazzino in data"
  La variabile vdata è assegnata da uno script.
- Due livelli di rottura
  1. Per ogni pittore, vogliamo una classificazione delle opere a magazzino.
     Si tratta del primo livello di rottura, che corrisponde alla zona R1-R2.
     Per questa zona scriviamo due script.
  2. All'interno di questo livello, vogliamo avere un secondo livello di rottura sulle date di pubblicazione delle opere di ogni pittore. Questo livello corrisponde alla zona R2-D.
     In questa zona assegnamo due script.

Ad ogni livello di rottura, 4th DIMENSION carica in memoria la prima scheda della selezione corrente e ciò vi consente di assegnare il contenuto di un campo per ogni zona di intestazione corrispondente ad una rottura.
- Le due intestazioni
  1. Inseriamo il riferimento del pittore. Questa zona di intestazione corrisponde a T-T1.
  2. Inseriamo il campo Anno. Questa zona di intestazione corrisponde a T1-T2.

Vogliamo infine stabilire un totale generale che sia la somma di tutti i sotto totali.
Avremo la rottura di livello 0, che corrisponde alla zona R0-R1, nella quale inseriamo anche due script.

Il vostro formato si presenta in questo modo:

Scrivete la procedura di chiamata del formato e gli script per il campo e le variabili associate.

```
IMPOSTA ARCHIVIO CORRENTE([Poster])
TUTTE LE SCHEDE
ORDINA SELEZIONE([Poster]Pittore;<;[Poster]Anno;>;[Poster]Titolo;>)
FORMATO LISTA("Stampe")
LIVELLI DI ROTTURA(2)
ACCUMULA SU([Poster]Quantità;[Poster]Valore)
STAMPA SELEZIONE
```

Script per le diverse variabili di formato

Quando eseguite la procedura avviando la stampa, il risultato si presenterà in questo modo:

RIPRODUZIONI A MAGAZZINO: Pag n°: 1

Riferimento: Van Gogh

Quadro(i) del:	1888	**Qtà**	**Valore**
Caffé di notte		2	560000
Sotto totale dell'anno :	1888	2	560000

Quadro(i) del:	1890	**Qtà**	**Valore**
Cipressi		9	955000
La chiesa di Auvers		5	560000
Sotto totale dell'anno :	1890	14	1515000

Sotto totale per: Van Gogh 16 2075000

Riferimento: Kandinsky

Quadro(i) del:	1910	**Qtà**	**Valore**
Acquarello astratto		3	785000
Sotto totale dell'anno :	1910	3	785000

Quadro(i) del:	1912	**Qtà**	**Valore**
Punte nell'arco		7	524000
Sotto totale dell'anno :	1912	7	524000

Quadro(i) del:	1913	**Qtà**	**Valore**
Composizione n.7		2	1200000
Sotto totale dell'anno :	1913	2	1200000

Sotto totale per: Kandinsky 12 2509000

Riferimento: Brueghel

Quadro(i) del:	1560	**Qtà**	**Valore**
Giochi di fanciulli		5	782000
Sotto totale dell'anno :	1560	5	782000

Quadro(i) del:	1565	**Qtà**	**Valore**
Raccolta del grano		8	782000
Sotto totale dell'anno :	1565	8	782000

Quadro(i) del:	1568	**Qtà**	**Valore**
Danza di contadini		3	1500000
Sotto totale dell'anno :	1568	3	1500000

Sotto totale per: Brueghel 16 3064000

Totale generale..: 44 7648000

Riferimenti

Somma, Livello

RAPPORTO REPORT

Sintassi
RAPPORTO({Archivio};{Documento};{*})

Parametri	**Tipo**	**Descrizione**
Archivio	Espr_Archivio	Archivio contenente la selezione da stampare
IlDocumento	Espr_Alfa	Nome del documento di definizione del rapporto
*	Costante	Richiesta di non visualizzazione delle finestre di dialogo per la stampa

Descrizione
RAPPORTO permette di riprodurre le funzioni della finestra di dialogo per la creazione dei rapporti che si trova nell'ambiente **Utilizzo**. Potrete creare un nuovo rapporto, richiedere l'apertura di un rapporto già registrato o proporre l'apertura di un altro rapporto di vostra scelta. *Archivio* è un argomento opzionale. Se viene omesso, RAPPORTO si riferirà all'archivio corrente.

- Se *IlDocumento* non è una stringa vuota e nessun documento ha questo nome, 4th DIMENSION vi propone la finestra di dialogo per la creazione di un nuovo rapporto. Se volete assicurarvi che questa finestra venga aperta, usate, ad esempio, la stringa Carattere(1). Infatti, questo carattere è inaccessibile da tastiera e non è quindi possibile che esista alcun documento con questo nome.

- Se *IlDocumento* non è una stringa vuota e corrisponde al nome di un documento esistente, RAPPORTO carica in memoria la descrizione del rapporto e lo stampa utilizzando i parametri di stampa scelti al momento della sua definizione.

- Se *IlDocumento* è una stringa vuota, RAPPORTO visualizzerà la finestra di dialogo standard per l'apertura dei documenti, con cui potrete scegliere il rapporto desiderato.

L'argomento * vi consente di evitare l'apertura delle finestre di dialogo per la stampa. 4th DIMENSION utilizzerà allora i formati di stampa impostati durante la creazione del rapporto.

Esempio

```
IMPOSTA ARCHIVIO CORRENTE([Clienti])
TUTTE LE SCHEDE                              Seleziona tutte le schede
DIALOGO("StampaRapporti")                    Chiedi il tipo di stampa che vogliamo realizzare
Caso fra
    :(pNuovo=1)
    RAPPORTO(Carattere(1))                   Crea un nuovo rapporto
    :(pStandard=1)
    RAPPORTO("Rapporto1")                    Apri e stampa "Rapporto1"
    :(pMio=1)
    RAPPORTO("")                             Apri un rapporto di mia scelta.
Fine caso
```

Riferimenti
STAMPA ETICHETTE, STAMPA SELEZIONE

PARAMETRI DI STAMPA

PRINT SETTINGS

Sintassi
PARAMETRI DI STAMPA

Descrizione
PARAMETRI DI STAMPA chiama le due finestre di dialogo standard per la stampa, quella per la scelta del formato e, successivamente, quella per avviare la stampa. Se fate clic su Annulla in una delle due finestre, la variabile di sistema OK assume il valore di 0, altrimenti OK assume il valore di 1. Questa procedura deve essere chiamata prima del comando STAMPA FORMATO, descritto qui di seguito. Non è di alcuna utilità chiamare PARAMETRI DI STAMPA prima di un comando STAMPA SELEZIONE o STAMPA ETICHETTE.

Finestre di dialogo per la stampa della stampante LaserWriter:

Finestre di dialogo per la stampa della stampante ImageWriter:

Formato di Stampa ImageWriter v2.7 OK
Carta: Lettera USA · A4 · Legale USA · Tabulato · Carta Computer — Annulla
Orientamento — Effetti Speciali: Verticale Grafico · Riduzione 50% · Pagine Continue

ImageWriter v2.7 OK
Qualità: Alta · Standard · Solo Testo
Pagine da Stampare: Tutte · Da: · A: — Annulla
Copie: 1
Alimentazione: Automatica · Manuale
Anteprima

Esempio
Fate riferimento all'esempio che descrive il comando STAMPA FORMATO

Riferimenti
STAMPA FORMATO

STAMPA FORMATO

PRINT LAYOUT

Sintassi
STAMPA FORMATO({Archivio};IlFormato)

Parametri	Tipo	Descrizione
Archivio	Espr_Archivio	Archivio al quale appartiene il formato da stampare
IlFormato	Espr_Alfa	Formato da utilizzare con STAMPA FORMATO

Descrizione
STAMPA FORMATO stampa il formato *IlFormato* dell'archivio *Archivio*. *Archivio* è opzionale. Se questo argomento viene omesso, STAMPA FORMATO cercherà il formato *IlFormato* nell'archivio corrente.
STAMPA FORMATO non chiama le finestre di dialogo per la stampa standard e stampa soltanto il dettaglio (compreso tra gli indicatori T e D) del formato *IlFormato*. Verranno eseguiti soltanto i comandi che si trovano nei cicli di Prima e di Durante, come pure gli script associati al dettaglio del formato.
Quando usate il comando STAMPA FORMATO, 4th DIMENSION costruisce in memoria la pagina da stampare, aspettando le chiamate di comandi successive e stamperà la pagina soltanto dopo che questa sarà stata completata o quando chiamerete il comando CAMBIO PAGINA, descritto qui di seguito.
Se CAMBIO PAGINA non viene chiamato, l'ultima pagina non completata non sarà stampata con le altre.

Esempio
Vogliamo stampare la selezione di un archivio di persone. Vogliamo cambiare il formato in funzione della stato civile della persona i cui dati sono in stampa.

Stato di famiglia del personale al: Giovedi 6 Aprile 1989

Signora Anna Leonardi
Coniugata Rondinelli
Nome del coniuge Oliviero
♀ *Età: 26 anni*

Signor Oliviero Rondinelli
Nome del coniuge Anna
♂ *Età: 27 anni*

Signorina Giovanna Lanzetti
♀ *Età: 25 anni*

Signor Sergio Minetti
♂ *Età: 29 anni*

Signor Mario Antonioli
♂ *Età: 29 anni*

Signora Giulia Dedini
Coniugata Vergnano
Nome del coniuge Enrico
♀ *Età: 23 anni*

Signorina Antonella Ceretta
♀ *Età: 26 anni*

Signorina Roberta Bonfiglioli
♀ *Età: 32 anni*

Suddivisione			
	♂	Coniugati	1
		Celibi	2
	♀	Coniugate	2
		Nubili	3

A questo scopo, creeremo un formato in base al tipo di stampa, senza dimenticare un formato di intestazione ed uno di pié di pagina, che ci darà il totale di ogni categoria.

La procedura di stampa è la seguente:

```
IMPOSTA ARCHIVIO CORRENTE([Persone])
PARAMETRI DI STAMPA                                    Chiama le finestre di dialogo per la stampa
TUTTE LE SCHEDE
VtotUs:=0                                              Inizializza le variabili di calcolo
VtotC:=0
VtotDs:=0
VtotN:=0
Vdata:=Data odierna
STAMPA FORMATO([Persone];"Intestazione")               Stampa l'intestazione della pagina
Mentre(Non(Fine selezione))                            Mentre non sei ancora alla fine dell'archivio
   Caso fra
      :([Persone]Genere="Uomo sposato")                Se Genere è uguale a Uomo sposato
      Se([Persone]Sposata)                             Se la persona è sposata
         STAMPA FORMATO([Persone];"Uomo sposato")      Stampa con il formato degli uomini sposati
            VtotUs:=VtotUs+1                           Accumula sul numero degli uomini sposati
      altrimenti                                       Se genere è uguale a Celibe
         STAMPA FORMATO([Persone];"Celibe")            Stampa con il formato degli uomini celibi
            VtotC:=VtotC+1                             Accumula sul numero degli uomini celibi
      Fine se
      :([Persone]Genere="Donna sposata")               Se Genere è uguale a Donna sposata
      Se([Persone]Sposata)                             Se la persona è sposata
      STAMPA FORMATO([Persone];"Donna sposata")        Stampa con il formato delle donne sposate
      VtotDs:=VtotDs+1                                 Accumula sul numero delle donne sposate
      altrimenti                                       Se il genere è uguale a Nubile
      STAMPA FORMATO([Persone];"Nubile")               Stampa con il formato delle donne nubili
      VtotN:=VtotN+1                                   Accumula sul numero delle donne nubili
      Fine se
   Fine caso
   SCHEDA SUCCESSIVA                                   Passa alla scheda successiva
Fine mentre
STAMPA FORMATO([Persone];"Pié di pagina")              Stampa il pié di pagina
CAMBIO PAGINA                                          Chiama CAMBIO PAGINA per finire la stampa
```

Riferimenti
PARAMETRI DI STAMPA, CAMBIO PAGINA

CAMBIO PAGINA — FORM FEED

Sintassi
CAMBIO PAGINA

Descrizione
CAMBIO PAGINA effettua la stampa della pagina preparata con STAMPA FORMATO e provoca un cambio pagina. Terminate sempre una stampa realizzata con STAMPA FORMATO chiamando CAMBIO PAGINA. Non occorre chiamare CAMBIO PAGINA nel caso di una stampa con STAMPA SELEZIONE.

Esempi e riferimenti :
STAMPA FORMATO, PARAMETRI DI STAMPA

Funzioni statistiche

Somma

Sum

Sintassi
Somma(IlCampo)
➤ Numerico

Parametri	Tipo	Descrizione
IlCampo	Espr_Campo	Campo in base al quale si vuole eseguire la somma

Contesto
Stampa, inserimento o procedura globale

Descrizione
Somma restituisce la somma dei valori di *IlCampo* per la selezione corrente o la sotto-selezione corrente. Questa funzione può essere usata durante la stampa o l'inserimento, per sommare i campi di un sotto-archivio, o su una selezione di un qualsiasi archivio della base dati. Attenzione: se volete eseguire la somma di una selezione di schede, 4th DIMENSION dovrà caricare tutte le schede e fare la somma dei valori, operazione che, ad esempio per 5000, richiederà un po' di tempo.

Esempio

vSomma:=**Somma**([Fatt]Righe'Totalenetto)	Somma di un sotto-campo
Se(In Piè di pagina)	
vTotale:=**Somma**([Fatt]Totalelordo)	Somma di un campo in stampa
Fine se	
TUTTE LE SCHEDE([Fatture])	
vSomma:=**Somma**([Fatt]Totalelordo)	Somma di un campo di una selezione di schede

Riferimenti
Media

Media

Average

Sintassi
Media(IlCampo)
➤ Numerico

Parametri	Tipo	Descrizione
IlCampo	Espr_Campo	Campo in base al quale si vuole effettuare una media

Descrizione
Media restituisce la media dei valori di *IlCampo* per la selezione o la sotto-selezione corrente. Questa funzione può essere usata in fase di stampa o di inserimento, per calcolare la media dei campi di un sotto-archivio, o su una selezione di un qualsiasi archivio della base dati. Se calcolate il valore medio di una selezione di schede, 4th DIMENSION dovrà caricare tutte le schede.

Esempio
Media di un sottocampo:

vMedia:=**Media**([Fatture]Righe'TotaleNetto)

Media di un campo in fase di stampa

Se (In piè di pagina)
    vMedia:=**Media**([Fatture]TerminiPagam)
Fine se

Media di un campo di una selezione di schede:

TUTTE LE SCHEDE([Fatture])
vMedia:=Media([Fatture]TotaleLordo)

Minimo — Min

Sintassi
Minimo(IlCampo)
➤ Numerico

Parametri	Tipo	Descrizione
IlCampo	Espr_Campo	Campo di cui si vuole conoscere il valore minimo

Descrizione
Minimo restituisce il valore minimo tra i valori di *IlCampo* per la selezione o la sotto-selezione corrente se *IlCampo* è un sotto-campo. Questa funzione può essere usata in fase di stampa o di inserimento, per calcolare il valore minimo dei campi di un sotto-archivio, o sulla selezione di un archivio. Se calcolate il valore minimo di una selezione di schede, 4th DIMENSION dovrà caricare tutte le schede.

Esempio
vMin:=**Min**([Fatture]Righe'Totalenetto) — Minimo di un sotto-campo

Se(In Pié di pagina)
    vMin:=**Min**([Regioni]Vendite)
Fine se

Minimo di un campo in fase di stampa

TUTTE LE SCHEDE([Clienti])
vMin:=**Min**([Clienti]Fatturato)

Minimo del campo di una selezione di schede

Riferimenti
Massimo

Massimo — Max

Sintassi
Massimo(IlCampo)
➤ Numerico

Parametri	Tipo	Descrizione
IlCampo	Espr_Campo	Campo di cui si vuole conoscere il valore massimo

Descrizione
Massimo restituisce il valore massimo tra i valori di *IlCampo* per la selezione o la sotto-selezione corrente se *IlCampo* è un sotto-campo. Questa funzione può essere usata in fase di stampa o di inserimento, per calcolare il valore massimo dei campi di un sotto-archivio, o sulla selezione di un archivio qualsiasi della base dati. Attenzione: se calcolate il valore massimo di una selezione di schede, 4th DIMENSION dovrà caricare tutte le schede e trovare il valore massimo, operazione che, per esempio per 5000 schede, richiederà un po' di tempo.

Esempio

```
vMax:=Max([Fatture]Righe'Totale netto)          Massimo di un sotto-campo

Se(In piè di pagina)                            Massimo di un campo in fase di stampa
    vMax:=Max([Regioni]Vendite)
Fine se

TUTTE LE SCHEDE([Clienti])                      Massimo del campo di una selezione di schede
vMax:=Max([Clienti]Fatturato)
```

Deviazione standard — Std deviation

Sintassi
Deviazione standard(IlCampo)
➤ Numerico

Parametri	Tipo	Descrizione
IlCampo	Espr_Campo	Campo di cui si vuole conoscere la deviazione standard

Contesto
Stampa

Descrizione
Deviazione standard restituisce la deviazione standard dei valori di *IlCampo*. Diversamente dalle funzio - ni Somma e Media, la funzione Deviazione standard non può essere riferita ad una selezione di schede, ma soltanto a due situazioni: nella zona di pié di pagina in fase di stampa e ad una sotto-selezione.

Somma dei quadrati — Sum squares

Sintassi
Somma dei quadrati(IlCampo)
➤ Numerico

Parametri	Tipo	Descrizione
IlCampo	Espr_Campo	Campo di cui si vuole conoscere la somma dei quadrati

Contesto
Stampa

Descrizione
Somma dei quadrati restituisce la somma dei quadrati dei valori di *IlCampo.*
Diversamente dalle funzioni Somma e Media, la funzione Somma dei quadrati non può essere riferita ad una selezione di schede, ma soltanto a due situazioni: nella zona di pié di pagina in fase di stampa e ad una sotto-selezione.

Riferimenti
Varianza, Deviazione standard

Varianza — Variance

Sintassi
Varianza(IlCampo)
➤ Numerico

Parametri	Tipo	Descrizione
IlCampo	Espr_Campo	Campo di cui i vuole conoscere la varianza

Contesto
Stampa (per i campi)

Descrizione
Varianza restituisce la varianza dei valori di *IlCampo.* Diversamente dalle funzioni Somma e Media, la funzione Varianza non può essere riferita ad una selezione di schede, ma soltanto a due situazioni: nella zona di pié di pagina in fase di stampa e ad una sotto-selezione.

ESPORTA SYLK

EXPORT SYLK

Sintassi
ESPORTA SYLK({Archivio};IlDocumento)

Parametri	**Tipo**	**Descrizione**
Archivio	Espr_Archivio	Archivio contenente la selezione da esportare
IlDocumento	Espr_Alfa	Nome del documento in cui viene esportata la selezione

Descrizione
ESPORTA SYLK esporta la selezione corrente dell'archivio *Archivio* usando il formato Lista corrente. I dati vengono scritti nel documento *IlDocumento*, utilizzando il formato SYLK. Si tratta di un formato standard per la trasmissione delle informazioni, riconosciuto, tra l'altro, da tutti i software Microsoft. Può darsi che questo documento esista già, oppure che sia nuovo. Nel primo caso, verrà cancellato da questa nuova esportazione. *IlDocumento* può contenere un percorso di accesso attraverso delle cartelle. Se *IlDocumento* è una stringa di caratteri vuota, ESPORTA SYLK vi proporrà la finestra standard per la creazione dei documenti. Potrete allora dare un nome al documento e scegliere il file di destinazione. In questo caso, potrete sapere se la registrazione è stata effettuata correttamente verificando la variabile OK, che assumerà il valore 1. Se avete annullato la registrazione, OK avrà invece il valore 0. La variabile *IlDocumento* conterrà il nome completo del documento creato.

ESPORTA SYLK esporta i campi e le variabili che si trovano nel formato Lista corrente, partendo da sinistra verso destra e dall'alto in basso. Inoltre, i cicli di Prima delle procedure di formato e degli script vengono eseguiti una volta per ogni scheda esportata.

Se non avete utilizzato il comando DISATTIVA MESSAGGIO, comparirà sullo schermo un indicatore di progressione. Se fate clic sul pulsante Interrompi, l'esportazione verrà annullata e la variabile OK assumerà il valore 0; diversamente, l'esportazione verrà portata a termine e la variabile OK assumerà il valore 1. Durante l'esportazione, 4th DIMENSION utilizza il filtro ASCII corrente. Se, ad esempio, volete filtrare i caratteri accentati, chiamate il comando USA FILTRO ASCII prima dell'esportazione (fate riferimento alla documentazione relativa ai filtri ASCII).

Esempio

```
IMPOSTA ARCHIVIO CORRENTE([Clienti])         Stiamo lavorando sull'archivio Clienti
TUTTE LE SCHEDE                              Seleziona tutte le schede
FORMATO LISTA("EsportClienti")               Seleziona il formato di esportazione
ESPORTA SYLK([Clienti];"ContiClienti.SYLK")  Esporta in SYLK i conti relativi ai clienti
```

Riferimenti
IMPORTA SYLK

ESPORTA DIF — EXPORT DIF

Sintassi
ESPORTA DIF({Archivio};IlDocumento)

Parametri	**Tipo**	**Descrizione**
Archivio	Espr_Archivio	Archivio contenente la selezione da esportare
IlDocumento	Espr_Alfa	Nome del documento in cui viene esportata la selezione

Descrizione
ESPORTA DIF esporta la selezione corrente dell'archivio *Archivio* utilizzando il formato Lista corrente. I dati vengono scritti nel documento *IlDocumento*. Può darsi che tale documento esista già oppure che sia nuovo. Nel primo caso, verrà cancellato da questa nuova esportazione. *IlDocumento* può contenere un percorso di accesso attraverso delle cartelle. Se *IlDocumento* è una stringa di caratteri vuota, ESPORTA DIF vi proporrà la finestra standard per la creazione dei documenti. Potrete allora dare un nome al documento e scegliere il file di destinazione. In questo caso, potrete sapere se la registrazione è stata effettuata correttamente verificando la variabile OK, che assumerà il valore 1. Se avete annullato la registrazione, OK avrà il valore 0. La variabile *IlDocumento* conterrà il nome completo del documento creato.

ESPORTA DIF esporta i campi e le variabili che si trovano nel formato Lista corrente, partendo da sinistra verso destra e dall'alto verso il basso. Inoltre, i cicli di Prima delle procedure di formato e degli script vengono eseguiti una volta per ogni scheda esportata.

Se non avete usato il comando CANCELLA MESSAGGI, comparirà sullo schermo un indicatore di progressione. Se fate clic sul pulsante Interrompi, l'esportazione verrà annullata e la variabile OK assumerà il valore 0; diversamente, l'esportazione giungerà a termine e la variabile OK assumerà il valore 1.
Durante l'esportazione, 4th DIMENSION usa il filtro ASCII corrente. Se, ad esempio, desiderate filtrare i caratteri accentati, chiamate il comando USA FILTRO ASCII prima dell'esportazione (fate riferimento alla documentazione relativa ai filtri ASCII).

Esempio

IMPOSTA ARCHIVIO CORRENTE([Clienti])	Stiamo lavorando sull'archivio Clienti
TUTTE LE SCHEDE	Seleziona tutte le schede
FORMATO LISTA("EsportClienti")	Seleziona il formato di esportazione
ESPORTA DIF([Clienti];ContiClienti.DIF")	Esporta in DIF i conti relativi ai clienti

Riferimenti
IMPORTA DIF

ESPORTA TESTO EXPORT TEXT

Sintassi
ESPORTA TESTO({Archivio};IlDocumento)

Parametri	**Tipo**	**Descrizione**
Archivio	Espr_Archivio	Archivio contenente la selezione da esportare
IlDocumento	Espr_Alfa	Nome del documento in cui viene esportata la selezione

Descrizione
ESPORTA TESTO esporta la selezione corrente dell'archivio *Archivio* utilizzando il formato Lista corrente. I dati vengono scritti in un documento chiamato *IlDocumento*. Può darsi che tale documento esista già oppure che sia nuovo. Nel primo caso, verrà cancellato da questa nuova esportazione. *IlDocumento* può contenere un percorso di accesso attraverso delle cartelle. Se *IlDocumento* è una stringa di caratteri vuota, ESPORTA TESTO vi proporrà la finestra standard per la creazione dei documenti. Potrete allora dare un nome al documento e scegliere il file di destinazione. In questo caso, potrete sapere se la registrazione è stata effettuata correttamente verificando la variabile OK, che assumerà il valore 1. Se avete annullato la registrazione, OK avrà il valore 0. La variabile *IlDocumento* conterrà il nome completo del documento creato.

ESPORTA TESTO esporta i campi e le variabili che si trovano nel formato Lista corrente, partendo da sinistra verso destra e dall'alto verso il basso. Inoltre, i cicli di Prima delle procedure di formato e degli script vengono eseguiti una volta per ogni scheda esportata.

Se non avete usato il comando CANCELLA MESSAGGI, comparirà sullo schermo un indicatore di progressione. Se fate clic sul pulsante Interrompi, l'esportazione verrà annullata e la variabile OK assumerà il valore 0; diversamente, l'esportazione giungerà a termine e la variabile OK assumerà il valore 1.
Durante l'esportazione, 4th DIMENSION usa il filtro ASCII corrente. Se, ad esempio, desiderate filtrare i caratteri accentati, chiamate il comando USA FILTRO ASCII prima dell'esportazione (fate riferimento alla documentazione relativa ai filtri ASCII).

ESPORTA TESTO esporta i dati utilizzando due separatori:
- Il separatore di campi: si tratta di un carattere inserito tra un campo e l'altro; se non viene specificato altrimenti, sarà inserito il tabulatore (carattere 9).
- Il separatore di schede viene invece inserito tra una scheda e l'altra; se non viene specificato altrimenti, sarà inserito il ritorno carrello (carattere 13). Potete modificare questi due separatori cambiando le due variabili di sistema FldDelimit (separatore di campo) e RecDelimit (separatore di scheda).

Attenzione
Se volete esportare delle schede contenenti aree di testo, è possibile che queste contengano dei caratteri di ritorno carrello. Non potrete quindi usare il ritorno carrello come separatore di schede. Se il software che dovrà rileggere questa informazione lo permette, potrete usare come separatori, ad esempio, i caratteri 201 e 203. Se volete esportare le aree di testo in Excel, ad esempio, usate la funzione Sostituisci stringa per sostituire i caratteri di ritorno carrello con degli spazi.

I dati esportati con il comando ESPORTA TESTO vengono esportati con un formato di lunghezza variabile. Se volete effettuare un'esportazione di lunghezza fissa, create un formato che contenga soltanto le variabili assegnate.
La procedura aggiunge gli spazi alla fine di IlCampo (20 rappresenta la lunghezza dell'area):

EsportVar1:=IlCampo+(" "*(20 - **Lunghezza**(IlCampo)))

Esempio

IMPOSTA ARCHIVIO CORRENTE([Clienti])	Stiamo lavorando sull'archivio Clienti
TUTTE LE SCHEDE	Seleziona tutte le schede
FORMATO LISTA("EsportClienti")	Seleziona il formato di esportazione
ESPORTA TESTO([Clienti];ContiClienti.Testo")	Esporta in ASCII i conti relativi ai clienti

Riferimenti
IMPORTA TESTO

IMPORTA SYLK — IMPORT SYLK

Sintassi
IMPORTA SYLK({Archivio};IlDocumento)

Parametri	**Tipo**	**Descrizione**
Archivio	Espr_Archivio	Archivio in cui viene effettuata l'importazione
IlDocumento	Espr_Alfa	Nome del documento da importare

Descrizione
IMPORTA SYLK importa i dati che si trovano nel documento *IlDocumento* in formato SYLK. I dati importati vengono registrati nelle schede dell'archivio *Archivio*.
IlDocumento può contenere un percorso di accesso attraverso delle cartelle. Se *IlDocumento* è una stringa di caratteri vuota, IMPORTA SYLK vi proporrà la finestra standard per la creazione dei documenti. Potrete allora scegliere il documento da importare. In questo caso, potrete conoscere il nome del documento importato servendovi della variabile di sistema Document. Se avete fatto clic sul pulsante Apri, la variabile OK assumerà il valore 1, altrimenti OK assumerà il valore 0.
IMPORTA SYLK importa i dati e li colloca nel formato Pagina corrente basandosi sull'ordine di inserimento definito nel formato, oppure da sinistra verso destra e dall'alto verso il basso, se non avete impostato nessun ordine di inserimento. Se il numero di campi o di variabili non corrisponde al numero di dati importati, 4th DIMENSION ignorerà tutti i valori supplementari. Durante l'importazione, i cicli di Dopo delle procedure di archivio, delle procedure di formato e degli script vengono eseguiti una volta per ogni scheda importata.
Se non avete usato il comando CANCELLA MESSAGGI, comparirà sullo schermo un indicatore di progressione. Se fate clic sul pulsante Interrompi, l'importazione verrà annullata e la variabile OK assumerà il valore 0; diversamente, l'esportazione verrà portata a termine e la variabile OK assumerà il valore 1.
Durante l'importazione, 4th DIMENSION usa il filtro ASCII corrente.

Riferimenti
ESPORTA SYLK

IMPORTA DIF

IMPORT DIF

Sintassi
IMPORTA DIF({Archivio};IlDocumento)

Parametri	Tipo	Descrizione
Archivio	Espr_Archivio	Archivio in cui viene effettuata l'importazione
IlDocumento	Espr_Alfa	Nome del documento da importare

Descrizione
MPORTA DIF importa i dati che si trovano nel documento *IlDocumento* in formato DIF. I dati così riletti vengono registrati nelle schede dell'archivio *Archivio. IlDocumento* può contenere un percorso di accesso attraverso delle cartelle. Se *IlDocumento* è una stringa di caratteri vuota, IMPORTA DIF vi proporrà la finestra standard per l'apertura dei documenti. Potrete allora scegliere il documento da importare. In questo caso, potrete sapere il nome del documento importato servendovi della variabile di sistema Document. Se fate clic sul pulsante Apri, la variabile OK assumerà il valore 1, altrimenti OK assumerà il valore 0.

IMPORTA DIF importa i dati e li colloca nel formato Pagina corrente basandosi sull'ordine di inserimento impostato nel formato, oppure da sinistra verso destra e dall'alto verso il basso se non avete definito nessun ordine di inserimento. Se il numero di campi o di variabili non corrisponde al numero di dati importati, 4th DIMENSION ignorerà tutti i valori supplementari. Durante l'importazione, i cicli di Dopo delle procedure di archivio, delle procedure di formato e degli script vengono eseguiti una volta per ogni scheda esportata.

Se non avete usato il comando DISATTIVA MESSAGGIO, sullo schermo comparirà un indicatore di progressione. Se fate clic sul pulsante Interrompi, l'importazione verrà annullata e la variabile OK assumerà il valore 0; diversamente, l'importazione giungerà a termine e la variabile OK assumerà il valore 1. Durante l'importazione, 4th DIMENSION usa il filtro ASCII corrente.

Riferimenti
ESPORTA DIF

IMPORTA TESTO   IMPORT TEXT

Sintassi
IMPORTA TESTO({Archivio};IlDocumento)

Parametri	Tipo	Descrizione
Archivio	Espr_Archivio	Archivio nel quale viene effettuata l'importazione
IlDocumento	Espr_Alfa	Nome del documento da importare

Descrizione
IMPORTA TESTO importa i dati che si trovano nel documento *IlDocumento* in formato ASCII.
I dati riletti vengono registrati nelle schede dell'archivio *Archivio*. *IlDocumento* può contenere un percorso di accesso attraverso delle cartelle. Se *IlDocumento* è una stringa di caratteri vuota, IMPORTA TESTO vi proporrà la finestra standard per l'apertura dei documenti. Potrete allora scegliere il documento da importare. In questo caso, potrete sapere il nome del documento importato utilizzando la variabile di sistema Document. Se fate clic sul pulsante Apri, la variabile OK assumerà il valore 1, altrimenti OK assumerà il valore 0.

IMPORTA TESTO importa i dati e li colloca nel formato Pagina corrente, basandosi sull'ordine di inserimento stabilito nel formato, oppure da sinistra verso destra e dall'alto verso il basso, se non avete definito nessun ordine di inserimento. Se il numero di campi o di variabili non corrisponde al numero di dati importati, 4th DIMENSION ignorerà tutti i valori supplementari. Durante l'importazione, i cicli di Dopo delle procedure di archivio, delle procedure di formato e degli script vengono eseguiti una volta per ogni scheda esportata.

Se non avete usato il comando DISATTIVA MESSAGGI, sullo schermo comparirà un indicatore di progressione. Se fate clic sul pulsante Interrompi, l'importazione verrà annullata e la variabile OK assumerà il valore 0; diversamente, l'importazione giungerà a termine e la variabile OK assumerà il valore 1. Durante l'importazione, 4th DIMENSION usa il filtro ASCII corrente.

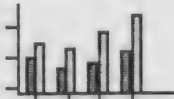
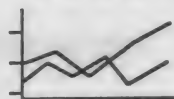
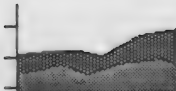
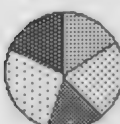
IMPORTA TESTO importa i dati servendosi di due separatori:
- Il separatore di campi: è un carattere introdotto tra un campo e l'altro; se non viene indicato altrimenti, sarà il tabulatore (carattere 9)
- Il separatore di schede è invece introdotto tra una scheda e l'altra; se non viene indicato altrimenti, sarà il carattere di ritorno carrello (carattere 13). Potete modificare questi due separatori cambiando le due variabili di sistema FldDelimit (separatore di campi) e RecDelimit (separatore di schede).

Attenzione a non usare lo stesso separatore contemporaneamente come separatore di schede e separatore di campi.

Riferimenti
ESPORTA TESTO

GRAFICO SELEZIONE

GRAPH FILE

Sintassi
GRAFICO SELEZIONE(Archivio{;Tipo;AsseX;AsseY1;...;AsseY8})

Parametri	Tipo	Descrizione
Archivio	Espr_Archivio	Archivio contenente la selezione su cui creare i grafici
Tipo	Espr_Num[1...8]	Tipo di grafico da creare
AsseX	Espr_Campo	Campo contenente i valori per l'asse delle x
AsseY1...AsseY8	Espr_Campo	Campo/i contenente/i i valori per l'asse delle Y

Descrizione
GRAFICO SELEZIONE è il comando equivalente al comando Grafici disponibile in ambiente Utilizzo. Se usate soltanto l'argomento *Archivio*, GRAFICO SELEZIONE vi presenterà la stessa finestra di dialogo disponibile in **Utilizzo**.
Se invece utilizzate anche gli argomenti *Tipo*, *AsseX*, *AsseY1...AsseY8,* il grafico sarà predeterminato tramite programmazione e non potrete cambiare i campi contenuti nel grafico. Potrete comunque utilizzare i comandi dei menu Disegni, Tipo di Grafico e Opzioni per modificare il tipo ed i parametri dei grafici.
Durante una chiamata di GRAFICO SELEZIONE, 4th DIMENSION sostituisce la barra di menu corrente con la barra di menu visualizzata in ambiente **Utilizzo** quando scegliete il comando **Grafici**.
GRAFICO SELEZIONE rappresenta sotto forma di grafico soltanto le prime 100 schede da voi selezionate.
I diversi tipi di grafico sono i seguenti:

Tipo di grafico	Tipo	Esempio
A Barre	1	
A Barre Proporzionali	2	
A Barre Sovrapposte	3	
A Linee	4	

Ad Area	5	
A Punti	6	
A Torta	7	
A Disegni	8	

Nota per la versione 1.0.x/versione 4.x.x
4th DIMENSION è ora in grado di accettare i campi di tipo numerico e data per l'asse delle X.

Esempio
Grafico del fatturato clienti:
IMPOSTA ARCHIVIO CORRENTE([Clienti])
TUTTE LE SCHEDE
GRAFICO SELEZIONE([Clienti];7;[Clienti]Società;[Clienti]Fatturato)

Grafico degli importi di fatturazione per date:
IMPOSTA ARCHIVIO CORRENTE([Clienti])
TUTTE LE SCHEDE
ORDINA SELEZIONE([Fatture];[Fatture]DataF;>)
GRAFICO SELEZIONE([Fatture];5;[Fatture]DataF;[Fatture]Importo)

Riferimenti
GRAFICO

GRAFICO — GRAPH

Sintassi
GRAFICO(IlGrafico;Tipo;AsseX;AsseY1;{...;AsseY8})

Parametri	Tipo	Descrizione
IlGrafico	Espr_Var	Nome della variabile grafico
Tipo	Espr_Num[1...8]	Tipo di grafico scelto
AsseX	Espr_Var o Espr_Vett o Espr_SottoArch	Valori per l'asse delle X
AsseY1...8	Espr_Var o Espr_Vett o Espr_SottoArch	Valori per gli assi Y1...Y8

Descrizione
GRAFICO vi permette di realizzare un grafico che verrà visualizzato in un formato per mezzo della variabile *IlGrafico* (consultate il *Manuale Ambiente Progetto* per sapere come creare queste variabili). I dati con cui il grafico viene creato possono appartenere ad un sotto-archivio o ad un vettore.
Diversamente dal comando GRAFICO SELEZIONE, GRAFICO non cambia la barra di menu corrente per sostituirla con quella dell'ambiente Utilizzo.
Non è quindi possibile modificare nessun parametro del grafico. Potete invece chiamare il comando GRAFICO nel ciclo di Durante delle vostre procedure di formato ed avere così la possibilità di modificare i parametri del grafico.
Tipo vi permette di cambiare il tipo di grafico. I valori utilizzati sono gli stessi validi per l'istruzione GRAFICO SELEZIONE.
AsseX può indicare sia i valori di un campo di tipo sotto-archivio, che un vettore di valori di tipo alfanumerico, numerico o data.
AsseY1;...;*AsseY8* possono indicare sia i valori di campi di sotto-strutture, che vettori di valori numerici.

Esempio
- Grafico su sotto-archivio

```
Se(Durante)
    TUTTE LE SOTTO_SCHEDE([Fatturato]Risultati)
    ORDINA SOTTO_SELEZIONE([Fatturato]Risultati'Mese;>)
    vTipo:=1
    GRAFICO(vGraf;vTipo;[Fatturato]Risultati'Mese;[Fatturato]Risultati'Vendite;[Fatturato]Risultati'Margine)
Fine se
```

- Grafico su vettori

```
Se(Durante)
    CERCA([Fatture]CodCliente=CodCliente)
    vtipo:=5
    SELEZIONE IN VETTORE([Fatture]Data;vettData;[Fatture]ImportoNetto;vettImportoNetto)
    GRAFICO(vGraf;vTipo;vettData;vettImportoNetto)
Fine se
```

Riferimenti
GRAFICO SELEZIONE

PARAMETRI GRAFICO — GRAPH SETTINGS

Sintassi
PARAMETRI GRAFICO(IlGrafico;MinX;MaxX;MinY;MaxY;XProp;GrigliaX;GrigliaY;{Titolo1...8})

Parametri	Tipo	Descrizione
IlGrafico	Espr_Var	Area grafica in un formato
MinX	Espr_Num o Espr_Data	Valore minimo per la scala delle X
MaxX	Espr_Num o Espr_Data	Valore massimo per la scala delle X
MinY	Espr_Num	Valore minimo per la scala delle Y
MaxY	Espr_Num	Valore massimo per la scala delle Y
xProp	Espr_Bool	Asse delle X proporzionale
GrigliaX	Espr_Bool	Griglia sull'asse delle X
GrigliaY	Espr_Bool	Griglia sull'asse delle Y
Titolo1...8	Espr_Alfa	Titoli da 1 a 8

Descrizione
PARAMETRI GRAFICO permette di riprodurre le funzioni accessibili nel menu Opzioni dell'ambiente Utilizzo, ossia di definire le scale e le griglie di un grafico. Questo comando si riferisce soltanto alle variabili associate alle zone grafiche che compaiono in un formato.

- *MinX* e *MaxX* rappresentano una coppia di valori di tipo numerico o data che permette di stabilire i valori minimo e massimo per l'asse delle X. Se *MinX* e *MaxX* sono uguali a 0, la scala delle X viene impostata automaticamente da 4th DIMENSION.

- *MinY* e *MaxY* rappresentano una coppia di valori di tipo numerico che permette di stabilire i valori minimo e massimo per l'asse delle Y. Se *MinY* e *MaxY* sono uguali a 0, la scala delle Y viene impostata automaticamente da 4th DIMENSION.

- *xProp* è un'espressione booleana che, se vera, imposta i valori sull'asse delle X in proporzione.

- *GrigliaX* è un'espressione booleana che, se vera, visualizza una griglia sull'asse delle X.

- *GrigliaY* è un'espressione booleana che, se vera, visualizza una griglia sull'asse delle Y.

Esempio

Stabiliamo i seguenti parametri per l'area grafica:
Scala delle X da 0 a 100, scala delle Y automatica
Asse delle X non proporzionale e griglie entrambi gli assi.

PARAMETRI GRAFICO(IlGrafico;0;100;0;0;Falso;Vero;Vero;"Vendite";"Risultato")

Riferimenti
GRAFICO

Insiemi

Generalità sugli insiemi

La selezione di un archivio rappresenta l'insieme delle schede trovate nel corso di una ricerca, oppure dopo aver selezionato un intero archivio.
Una selezione non contiene "fisicamente" le schede, ma soltanto i loro riferimenti.
Di un archivio viene caricata in memoria soltanto la scheda corrente.
Quando lavorate su un archivio, operate sempre con le schede della selezione.

Può essere utile, in certi casi, conservare queste selezioni, come ad esempio, conservare in memoria i diversi risultati delle ricerche per poterli successivamente combinare. L'uso degli *insiemi* vi dà questa possibilità.
Un insieme è un oggetto in memoria che create partendo da una selezione o da insiemi già esistenti.
Un insieme può anche essere vuoto.
Quando create un insieme, assegnategli un nome, ad esempio "Le Mie Schede".
L'insieme contiene le informazioni necessarie per sapere se gli appartiene o meno ogni scheda dell'archivio a cui è associato.

Selezione ed insieme sono due oggetti di **diversa natura**.
- La selezione è la lista delle schede di un archivio che state elaborando
- L'insieme è un oggetto che vi permette di conservare un gruppo di schede.
4th DIMENSION fornisce i comandi necessari per ottenere un insieme partendo dalla selezione di un archivio e di ottenere la nuova selezione di un archivio partendo da un insieme.

Quando create un insieme, grazie a questa operazione, lo associate sempre ad un archivio. Le operazioni che potete compiere sugli insiemi si possono effettuare soltanto su insiemi associati allo **stesso** archivio.

Nota tecnica
La dimensione di un insieme, espressa in byte, è sempre uguale al numero totale di schede contenute nell'archivio associato diviso per 8. Se definite un insieme associato ad un archivio contenente 10000 schede, l'insieme occuperà 1250 byte, cioè ≈1,2K di memoria centrale.
Non dovete quindi preoccuparvi della quantità di memoria occupata dai vostri insiemi, perchè questi ultimi non potranno provocare una saturazione della memoria centrale del vostro Macintosh.

Esempio di uso degli insiemi

Consideriamo il seguente archivio [Persone]

Dopo l'inserimento di un certo numero di schede, vogliamo verificare l'eventuale presenza di doppioni nell'archivio ed eliminarli con la stessa operazione.
Consideriamo il seguente archivio:

IMPOSTA ARCHIVIO CORRENTE([Persone])
TUTTE LE SCHEDE
vCognome:=""
vNome:=""
vGenere:=""
ORDINA SELEZIONE([Persone]Cognome;>;[Persone]Nome;>[Persone]Genere;>)
NUOVO INSIEME VUOTO("Doppioni")
APPLICA ALLA SELEZIONE(*CercaDoppioni*)
USA INSIEME("Doppioni")
CANCELLA SELEZIONE
ELIMINA INSIEME("Doppioni")

Inizializza le variabili vCognome, vNome e vGenere, che serviranno a conservare il cognome, il nome ed il genere della scheda precedente per poterne confrontare i valori. Ordina l'archivio in base a cognome, nome e genere in ordine ascendente.
Associa all'archivio l'insieme vuoto "Doppioni". All'uscita dal ciclo, l'insieme "Doppioni" contiene i doppioni.
Crea la nuova selezione dell'archivio a partire da questo insieme. Elimina le schede della selezione. Cancella l'insieme e le variabili diventate inutili.

Procedura *CercaDoppioni*

Se([Persone]Cognome=vCognome)&([Persone]Nome=vNome)&([Persone]Genere=vGenere)
    AGGIUNGI AD INSIEME("Doppioni")
altrimenti
    vGenere:=[Persone]Genere
    vNome:=[Persone]Nome
    vCognome:=[Persone]Cognome
Fine se

Al termine della procedura, l'archivio si presenta come segue:

Persone : 5 su 5

Genere	Cognome	Nome
Signor	DEANI	Paolo
Signor	DEANI	Oreste
Signora	MIGLIORINI	Isabella
Signorina	MIGLIORINI	Isabella
Signorina	MIGLIORINI	Ornella

NUOVO INSIEME

CREATE SET

Sintassi
NUOVO INSIEME({Archivio};Insieme)

Parametri	**Tipo**	**Descrizione**
Archivio	Espr_Archivio	Archivio contenente la selezione
Insieme	Espr_Alfa	Nome dell'insieme

Descrizione
NUOVO INSIEME crea un insieme chiamato *Insieme* contenente le schede della selezione corrente dell'archivio *Archivio*.
Archivio è un argomento opzionale. Se tale argomento viene omesso, NUOVO INSIEME si riferirà alla selezione corrente dell'archivio corrente.
Questo insieme viene creato nella memoria centrale ed occupa 1 bit per ogni scheda. Un insieme non può essere conservato ordinato. L'informazione relativa alla scheda corrente di *Archivio* viene conservata con l'insieme.
Se l'insieme esiste già, NUOVO INSIEME cancella l'insieme precedente e lo sostituisce con quello nuovo.

Esempio

IMPOSTA ARCHIVIO CORRENTE([Clienti])	Stiamo lavorando sull'archivio Clienti
CERCA([Clienti]DaPagare>0)	Cerca i clienti che hanno un saldo da pagare superiore a 0
Se(**Schede in selezione**>0)	Se ce ne sono
NUOVO INSIEME("Dasollecitare")	Registra questa selezione nell'insieme "Dasollecitare", perché possa essere usata successivamente.
Fine se	

Riferimenti
NUOVO INSIEME VUOTO

NUOVO INSIEME VUOTO — CREATE EMPTY SET

Sintassi
NUOVO INSIEME VUOTO({Archivio};Insieme)

Parametri	**Tipo**	**Descrizione**
Archivio	Espr_Archivio	Archivio dell'insieme da creare
Insieme	Espr_Alfa	Nome dell'insieme

Descrizione
NUOVO INSIEME VUOTO crea un insieme vuoto chiamato *Insieme*, collegato all'archivio *Archivio*. *Archivio* è un argomento opzionale. Se viene omesso, NUOVO INSIEME VUOTO si riferirà all'archivio corrente. Potrete inserire in seguito delle schede, utilizzando il comando AGGIUNGI AD INSIEME.

Esempio

IMPOSTA ARCHIVIO CORRENTE([Clienti])	Stiamo lavorando sull'archivio Clienti
NUOVO INSIEME VUOTO("NuoveSchede")	Crea un insieme vuoto chiamato "NuoveSchede"
Ripeti	Ripeti la seguente sequenza
AGGIUNGI SCHEDA	Aggiungi un cliente
Se(OK=1)	Se l'inserimento è confermato
AGGIUNGI AD INSIEME("NuoveSchede")	Inserisci la scheda nell'insieme "NuoveSchede"
Fine se	La sequenza viene ripetuta finché l'inserimento non è
Finché(OK=0)	terminato.

Riferimenti
NUOVO INSIEME, AGGIUNGI AD INSIEME

AGGIUNGI AD INSIEME — ADD TO SET

Sintassi
AGGIUNGI AD INSIEME({Archivio};Insieme)

Parametri	**Tipo**	**Descrizione**
Archivio	Espr_Archivio	Archivio contenente la scheda corrente da inserire
Insieme	Espr_Alfa	Nome dell'insieme

Descrizione
AGGIUNGI AD INSIEME aggiunge la scheda corrente dell'archivio *Archivio* all'insieme *Insieme*. L'archivio è un argomento opzionale. Se tale argomento viene omesso, AGGIUNGI AD INSIEME si riferirà all'archivio corrente. Se la scheda corrente si trova già nell'insieme *Insieme*, AGGIUNGI AD INSIEME non esegue alcuna azione.

Per l'esempio ed i riferimenti consultate la descrizione del comando successivo.

USA INSIEME

USE SET

Sintassi
USA INSIEME(Insieme)

Parametri	Tipo	Descrizione
Insieme	Espr_Alfa	Nome dell'insieme da utilizzare

Descrizione
USA INSIEME rende le schede presenti in Insieme la selezione corrente dell'archivio a cui questo insieme appartiene. La scheda corrente sarà quella registrata in *Insieme*.

Esempio

```
IMPOSTA ARCHIVIO CORRENTE([Clienti])           Stiamo lavorando sull'archivio Clienti
NUOVO INSIEME VUOTO ("NuoveSchede")            Crea l'insieme vuoto "NuoveSchede"
Ripeti                                         Ripeti la seguente sequenza
    AGGIUNGI SCHEDA                            Aggiungi un cliente
    Se(OK=1)                                   Se l'inserimento è stato confermato
        AGGIUNGI AD INSIEME("NuoveSchede")     Inserisci la scheda in "NuoveSchede"
    Fine se
Finché(OK=0)                                   Sequenza ripetuta sino alla fine dell'inserimento
CONFERMA("Vuoi stampare le schede inserite?")  Chiedi conferma per la stampa delle schede
Se(OK=1)                                       Se viene fatto clic su OK
    USA INSIEME("NuoveSchede")                 Trasforma l'insieme in selezione corrente
    STAMPA SELEZIONE                           Stampa le schede
Fine se
```

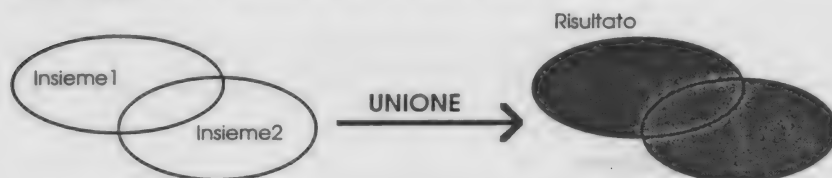
Riferimenti
NUOVO INSIEME, UNIONE, DIFFERENZA, INTERSEZIONE

UNIONE

UNION

Sintassi
UNIONE(Insieme1;Insieme;Risultato)

Parametri	Tipo	Descrizione
Insieme1	Espr_Alfa	Primo insieme
Insieme2	Espr_Alfa	Secondo insieme
Risultato	Espr_Alfa	Insieme generato dall'unione

Descrizione
UNIONE esegue l'unione degli insiemi *Insieme1* ed *Insieme2* e registra il risultato di questa operazione in *Risultato*. *Risultato* conterrà tutte le schede di *Insieme1* più quelle di *Insieme2*. Ovviamente, i due insiemi di partenza devono appartenere allo stesso archivio.

Dopo l'unione, se chiamate il comando USA INSIEME, la scheda corrente sarà quella registrata in *Insieme1*.

L'insieme *Risultato* può essere uno degli insiemi *Insieme1* o *Insieme2,* ciò significa che potete scrivere: **UNIONE**(Insieme1;Insieme2;Insieme1). *Insieme1* verrà quindi modificato per accogliere il risultato dell'unione dei due insiemi di partenza.

Esempio

UNIONE("AbitaTorino";"AbitaMilano";"DaStampare")	Insieme comprendente gli abitanti di Torino e di Milano

Riferimenti
INTERSEZIONE, DIFFERENZA

INTERSEZIONE — INTERSECTION

Sintassi
INTERSEZIONE(Insieme1;Insieme2;Risultato)

Parametri	**Tipo**	**Descrizione**
Insieme1	Espr_Alfa	Primo insieme
Insieme2	Espr_Alfa	Secondo insieme
Risultato	Espr_Alfa	Insieme derivante dall'intersezione

Descrizione
INTERSEZIONE esegue l'intersezione degli insiemi *Insieme1* ed *Insieme2* e registra l'insieme ottenuto in *Risultato*, che conterrà unicamente le schede che appartenevano contemporaneamente ad *Insieme1* e *Insieme2*. Ovviamente, i due insiemi devono appartenere allo stesso archivio.

Se chiamate il comando USA INSIEME dopo l'intersezione, la scheda corrente sarà la quella precedentemente registrata in Insieme1, se quest'ultima appartiene all'insieme *Risultato*.

Esempio

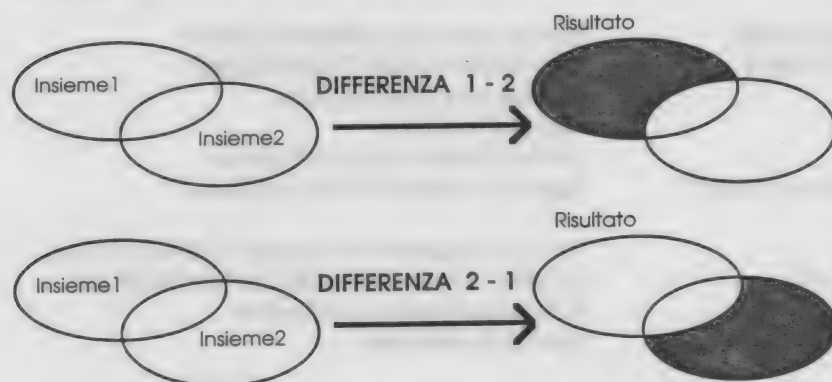
INTERSEZIONE("DaSollecitare";"AbitaParigi";"DaStampare")	Insieme contenente i clienti da sollecitare che abitano a Parigi.

Riferimenti
UNIONE, DIFFERENZA

DIFFERENZA — DIFFERENCE

Sintassi
DIFFERENZA(Insieme1;Insieme2;Risultato)

Parametri	Tipo	Descrizione
Insieme1	Espr_Alfa	Primo insieme
Insieme2	Espr_Alfa	Secondo insieme
Risultato	Espr_Alfa	Insieme derivante dall'intersezione

Descrizione
DIFFERENZA esegue la differenza tra *Insieme1* e *Insieme2,* registrando il risultato che ne deriva in *Risultato.* Attenzione: DIFFERENZA, diversamente da UNIONE e INTERSEZIONE, non è un'operazione commutativa, ciò significa che la differenza tra *Insieme1* ed *Insieme2* non è uguale alla differenza tra *Insieme2* ed *Insieme1*. Ovviamente, i due insiemi devono appartenere allo stesso archivio.

Se chiamate il comando USA INSIEME dopo un'operazione di differenza tra due insiemi, la scheda corrente sarà quella precedentemente registrata in *Insieme1,* se questa appartiene all'insieme *Risultato.*

Esempio

Esamineremo ora la stessa operazione, invertendo *Insieme1* ed *Insieme2*, per dimostrare che i risultati sono molto diversi.

DIFFERENZA("DaSollecitare";"AbitaParigi;"DaStampare")	Insieme contenente i clienti da sollecitare che non abitano a Parigi
DIFFERENZA("AbitaParigi";"DaSollecitare";"DaStampare")	Insieme contenente i clienti che abitano a Parigi e che non sono da sollecitare

Riferimenti

UNIONE, INTERSEZIONE

Schede in insieme — Records in set

Sintassi

Schede in insieme(Insieme)

➤ Numerico

Parametri	Tipo	Descrizione
Insieme	Espr_Alfa	Nome dell'insieme

Descrizione

Schede in insieme è una funzione che restituisce il numero di elementi contenuti in *Insieme*.

Esempio

```
IMPOSTA ARCHIVIO CORRENTE([Clienti])              Stiamo lavorando sull'archivio Clienti
NUOVO INSIEME VUOTO("NuoveSchede")                Crea l'insieme vuoto "NuoveSchede"
Ripeti                                            Ripeti la seguente sequenza
   AGGIUNGI SCHEDA                                Aggiungi un cliente
   Se(OK=1)                                       Se l'inserimento viene confermato
      AGGIUNGI AD INSIEME("NuoveSchede")          Aggiungi la scheda in "NuoveSchede"
   Fine se
Finché(OK=0)                                      Ripeti la sequenza fino al termine dell'inserimento
Se(Schede in insieme("NuoveSchede")>0)            Se esiste almeno una scheda in "NuoveSchede"
   USA INSIEME("NuoveSchede")                     Trasforma l'insieme in selezione
   STAMPA SELEZIONE                               Stampa la selezione
Fine se
```

Riferimenti

NUOVO INSIEME, UNIONE, INTERSEZIONE, DIFFERENZA

Appartiene ad insieme — Is in set

Sintassi
Appartiene ad insieme({Archivio};Insieme)
➤ Booleano

Parametri	Tipo	Descrizione
Archivio	Espr_Archivio	Archivio a cui è associato Insieme
Insieme	Espr_Alfa	Insieme in cui verificare

Descrizione
Appartiene ad insieme è una funzione che restituisce un valore vero se la scheda corrente di *Archivio* appartiene ad *Insieme*. *Archivio* è un argomento opzionale. Se questo argomento viene omesso, Appartiene ad insieme si riferirà all'archivio corrente.

Esempio
Immaginate di aver creato un insieme di clienti e che vogliate impedire ad alcuni utenti di accedere a questo insieme.

```
IMPOSTA ARCHIVIO CORRENTE([Clienti])
x:=Richiedi("Nome del cliente?")                  Richiedi il nome del cliente da modificare
Se(OK=1)                                          Se la richiesta è confermata
    CERCA([Clienti]Nome=x)                        Cerca la scheda
    Se(Schede in selezione=1)                     Se trovi la scheda, verifica se questa ap-
        Se(Appartiene ad insieme("AccessoRiservato"))  partiene all'insieme "AccessoRiservato"
            AVVISO("Non puoi modificare questa scheda")  Avvisa: non si può modificare la scheda
        altrimenti                                Se la scheda non appartiene all'insieme
            MODIFICA SCHEDA                       con accesso riservato, modifica la scheda
        Fine se
    Fine se
Fine se
```

Riferimenti
NUOVO INSIEME

ELIMINA INSIEME — CLEAR SET

Sintassi
ELIMINA INSIEME(Insieme)

Parametri	Tipo	Descrizione
Insieme	Espr_Alfa	Nome dell'insieme da eliminare

Descrizione

ELIMINA INSIEME cancella l'insieme *Insieme* dalla memoria centrale del Macintosh. Questo comando non cancella le schede che appartengono all'insieme, ma soltanto Insieme. Prendete l'abitudine di cancellare gli insiemi che non vi servono più.

Esempio

Riprendiamo l'esempio precedente

```
IMPOSTA ARCHIVIO CORRENTE([Clienti])
NUOVO INSIEME VUOTO("NuoveSchede")
Ripeti
   AGGIUNGI SCHEDA
   Se(OK=1)
      AGGIUNGI AD INSIEME("NuoveSchede")
   Fine se
Finché(OK=0)
Se(Schede in insieme("NuoveSchede")>0)
   USA INSIEME("NuoveSchede")
   STAMPA SELEZIONE
   ELIMINA INSIEME("NuoveSchede")
Fine se
```

Riferimenti

NUOVO INSIEME

REGISTRA INSIEME — SAVE SET

Sintassi

REGISTRA INSIEME(Insieme;NomeDoc)

Parametri	Tipo	Descrizione
Insieme	Espr_Alfa	nsieme da registrare
NomeDoc	Espr_Alfa	Nome del documento su disco

Descrizione

REGISTRA INSIEME registra l'insieme *Insieme* sul disco Macintosh nel documento *NomeDoc*. Se *NomeDoc* è una stringa di caratteri vuota, 4th DIMENSION vi presenterà la finestra standard per la creazione dei documenti. In questo caso, potrete sapere se la registrazione dell'insieme è stata confermata verificando la variabile OK, che assumerà il valore 1. La variabile di sistema Document conterrà il nome del documento su disco.

Attenzione:

Gli insiemi registrati non conservano la coerenza dei dati se avete cancellato ed aggiunto delle schede. In effetti, in questo caso alcune delle schede aggiunte potrebbero appartenere all'insieme caricato, mentre non esistevano al momento della creazione dell'insieme.

In questo caso non dovrete usare i comandi REGISTRA INSIEME e CARICA INSIEME per registrare una ricerca.

Esempio

REGISTRA INSIEME("Sollecitare";"Sollecita.Insieme")	Registra l'insieme Sollecitare nel documento Sollecita.Insieme

Riferimenti
CARICA INSIEME

CARICA INSIEME — LOAD SET

Sintassi
CARICA INSIEME(Archivio;Insieme;NomeDoc)

Parametri	Tipo	Descrizione
Archivio	Espr_Archivio	Nome dell'archivio a cui associare l'insieme
Insieme	Espr_Alfa	Nome dell'insieme da creare
NomeDoc	Espr_Alfa	Nome del documento di tipo insieme registrato su disco

Descrizione
CARICA INSIEME crea in memoria un insieme chiamato Insieme, associato all'archivio *Archivio*, rileggendolo dal documento Macintosh *NomeDoc*. Se il documento non esiste, la variabile OK assume il valore 0, altrimenti il valore sarà 1. Se *NomeDoc* è una stringa vuota, 4th DIMENSION vi proporrà la finestra di dialogo standard per l'apertura dei documenti. Potete conoscere il nome del documento aperto usando la variabile Document.

Esempio

CARICA INSIEME([Clienti];"DaSollecitare";"")	Carica un documento di solleciti NomeDoc è una stringa vuota Scegli un documento
CARICA INSIEME([Clienti];"DaSollecitare";"Sollecita.Insieme")	Carica il documento di solleciti Sollecita.Insieme

Riferimenti
REGISTRA INSIEME

Messaggi

AVVISO

ALERT

Sintassi
AVVISO(Messaggio)

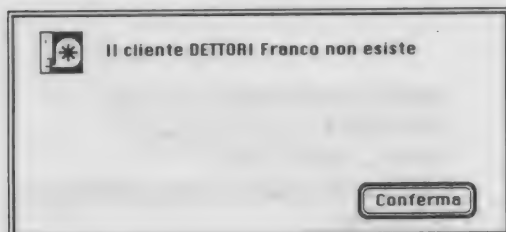
Parametri	Tipo	Descrizione
Messaggio	Espr_Alfa	Messaggio da visualizzare nella finestra di dialogo (max 80 caratteri)

Descrizione
AVVISO visualizza sullo schermo una finestra, contenente *Messaggio*. Chiamate AVVISO quando volete segnalare un errore o un'azione che non si è svolta correttamente. Il messaggio deve essere un'espressione alfanumerica. Se volete visualizzare valori di tipo numerico o data, usate la funzione Stringa. Se volete inserire un ritorno carrello all'interno del messaggio, usate la funzione Carattere(13); per inserire invece le virgolette, usate la funzione Carattere(34).

Esempio

AVVISO("Ho trovato "+**Stringa**(**Schede in selezione**)+" schede corrispondenti alla ricerca")

AVVISO("Buongiorno "+**Carattere**(13)+"E' il "+**Stringa**(**Data odierna**))

AVVISO("Il cliente "+vCognome+" "+vNome+" non esiste")

Riferimenti
CONVALIDA, Richiedi

CONVALIDA

CONFIRM

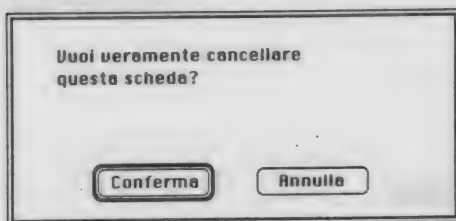
Sintassi
CONVALIDA(Messaggio)

Parametri	Tipo	Descrizione
Messaggio	Espr_Alfa	Messaggio di conferma da visualizzare

Descrizione

CONVALIDA apre una finestra di dialogo di conferma contenente due pulsanti, Conferma e Annulla, insieme alla stringa di caratteri *Messaggio*. Se fate clic su Conferma o premete il tasto A-capo, la variabile di sistema OK assumerà il valore 1, altrimenti OK avrà il valore 0. Usate questo comando per richiedere la conferma di un'azione.

Esempio

```
CONVALIDA("Vuoi veramente eliminare questa scheda?")
Se(OK=1)
    CANCELLA SCHEDA
Fine se
```

Riferimenti

AVVISO, Richiedi

Richiedi — Request

Sintassi

Richiedi(Messaggio;{ValoreCorrente})
➤ Stringa

Parametri	Tipo	Descrizione
Messaggio	Espr_Alfa	Messaggio che richiede di inserire un valore
ValoreCorrente	Espr_Alfa	Valore predeterminato da visualizzare nella finestra

Descrizione

Richiedi apre una finestra di dialogo contenente la stringa di caratteri *Messaggio*, un'area di testo inseribile contenente *ValoreCorrente* e 2 pulsanti, Conferma e Annulla. Se fate clic su Conferma o se premete il tasto A-capo, la variabile di sistema OK assumerà il valore 1 e Richiedi restituirà il valore inserito nell'area di testo. Se fate clic su Annulla, OK avrà il valore 0 e Richiedi restituirà una stringa vuota.

Esempio

```
LaStringa:=Richiedi("Indica il numero del mese:";"11")
vMese:=Num(LaStringa)
Se(OK=1)&(vMese#0)
    CERCA CON INDICE([Fatture]Mese=vMese)
    ...............
Fine se
```

Recupera il risultato nella variabile numerica vMese
Se la richiesta è stata confermata e se il valore inserito è diverso da 0, cerca le fatture corrispondenti.

Riferimenti
AVVISO, CONVALIDA

MESSAGGIO — MESSAGE

Sintassi
MESSAGGIO(Messaggio)

Parametri	Tipo	Descrizione
Messaggio	Espr_Alfa	Messaggio da visualizzare

Descrizione
1/ Se nessuna delle finestre create con il comando NUOVA FINESTRA è aperta:
4th DIMENSION apre una finestra e visualizza *Messaggio*. Ad ogni chiamata del comando MESSAGGIO, 4th DIMENSION cancella il contenuto della finestra prima di visualizzare il nuovo messaggio

2/ Se sullo schermo c'è una finestra aperta con il comando NUOVA FINESTRA:
4th DIMENSION visualizza *Messaggio* nel punto in cui si trova il cursore. Ad ogni chiamata del comando MESSAGGIO, 4th DIMENSION, diversamente dal primo caso, non cancella il contenuto della finestra, ma colloca Messaggio nella posizione attuale del cursore, cioé quella successiva all'ultimo inserimento, oppure in un punto qualsiasi se avete usato il comando VAI A XY (riferitevi al capitolo Finestre).

Note relative ai caratteri utilizzati:
In una finestra creata con il comando NUOVA FINESTRA, i messaggi sono visualizzati in Monaco 9 punti. Non è possibile cambiare il tipo di carattere. In compenso, questo tipo di carattere non è proporzionale (tutti i caratteri presentano la stessa larghezza) e permette così di allineare facilmente il testo in colonne.

Esempio

```
IMPOSTA ARCHIVIO CORRENTE([Clienti])
TUTTE LE SCHEDE                                        Per tutte le schede
Ciclo(i;1;Schede in selezione)                         Visualizza il nome del cliente in corso
   MESSAGGIO("Aggiornamento del cliente "+[Cliente]Nome)   di aggiornamento
   ....
   SCHEDA SUCCESSIVA
Fine ciclo
```

Riferimenti
AVVISO, NUOVA FINESTRA, VAI A XY

DISATTIVA MESSAGGI — MESSAGES OFF

Sintassi
DISATTIVA MESSAGGI

Descrizione
DISATTIVA MESSAGGI fa sì che non venga visualizzata la finestra dell'indicatore di progressione durante gli ordinamenti, le ricerche sequenziali, le chiamate del comando APPLICA ALLA SELEZIONE, le importazioni e le esportazioni di dati. Questo comando rimane attivo finché non chiamate il comando ATTIVA MESSAGGI, descritto qui di seguito, e non elimina la visualizzazione dei messaggi d'errore. Se volete controllare la visualizzazione dei messaggi d'errore, usate il comando CHIAMA SU ERRORE.

Esempio

```
IMPOSTA ARCHIVIO CORRENTE([Clienti])
TUTTE LE SCHEDE
DISATTIVA MESSAGGI
ORDINA SELEZIONE([Clienti]Società;>)
ATTIVA MESSAGGI
.....` Seguito della vostra procedura
```

Riferimenti
ATTIVA MESSAGGI

ATTIVA MESSAGGI — MESSAGES ON

Sintassi
ATTIVA MESSAGGI

Descrizione
ATTIVA MESSAGGI ripristina la visualizzazione della finestra dell'indicatore di progressione durante gli ordinamenti, le ricerche sequenziali, le chiamate del comando APPLICA ALLA SELEZIONE, le importazioni e le esportazioni di dati, quando la visualizzazione è stata eliminata con l'uso del comando DISATTIVA MESSAGGI.

Esempio
Fate riferimento all'esempio di DISATTIVA MESSAGGI

Riferimenti
DISATTIVA MESSAGGI

MOSTRA SCHEDA — DISPLAY RECORD

Sintassi
MOSTRA SCHEDA({Archivio})

Parametri	Tipo	Descrizione
Archivio	Espr_Archivio	Archivio contenente la scheda da visualizzare

Descrizione
MOSTRA SCHEDA visualizza la scheda corrente dell'archivio *Archivio* utilizzando il formato Pagina corrente, finchè un altro comando non interviene a modificare lo schermo. *Archivio* è opzionale. Se questo argomento viene omesso, MOSTRA SCHEDA si riferirà all'archivio corrente.

Esempio
Potete realizzare una presentazione con 4th DIMENSION. Procedete in questo modo:

```
IMPOSTA ARCHIVIO CORRENTE([Pres])
TUTTE LE SCHEDE                            Seleziona tutte le schede
InterrompiPres:=0                          Inizializza la variabile di interruzione
Tempo:=Num(Richiedi("Temporizzazione"))    Chiedi la temporizzazione desiderata
CHIAMA SU EVENTO("InterrompiPres")         Installa la procedura d'interruzione
Ripeti
MOSTRA SCHEDA                              Visualizza la scheda
Aspetta(Tempo)                             Attendi per una durata uguale a Tempo
SCHEDA SUCCESSIVA                          Passa alla scheda successiva
Se(Fine selezione)                         Se arrivi alla fine della selezione
PRIMA SCHEDA                               Ritorna all'inizio della selezione
Fine se
Finché(InterrompiPres=1)                   Ripeti l'operazione finchè InterrompiPres non sarà
CHIAMA SU EVENTO("")                       uguale a 1
                                           Termina e disattiva la procedura d'interruzione
```

Procedura d'interruzione InterrompiPres

```
InterrompiPres:=1                          Al primo evento, interrompi la presentazione
```

ProceduraAspetta

```
$oraInizio:=Ora attuale                    Considera l'ora attuale
Mentre(Ora attuale-$oraInizio>$1)          Finchè il termine ricevuto non è trascorso
Fine mentre                                Ritorna nel ciclo
```

Riferimenti
VISUALIZZA SELEZIONE, MESSAGGIO

Sostituisci caratteri

Change string

Sintassi

Sostituisci caratteri(Sorgente;Sostituzione;Posizione)

➤ Stringa

Parametri	**Tipo**	**Descrizione**
Sorgente	Espr_Alfa	Stringa di partenza
Sostituzione	Espr_Alfa	Stringa da mettere in Sorgente
Posizione	Espr_NumPos	Posizione di partenza della sostituzione

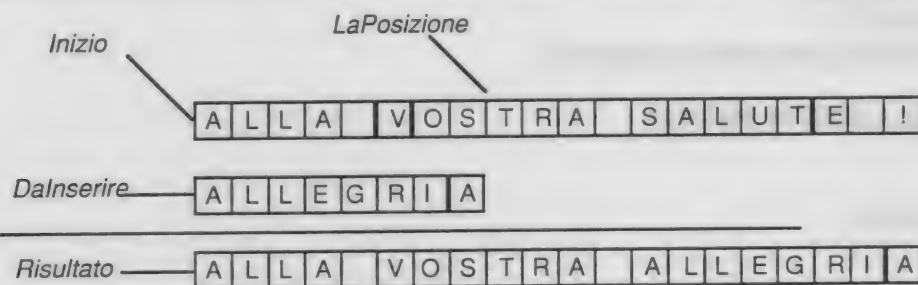
Descrizione

Sostituisci caratteri restituisce una stringa ottenuta prendendo la stringa *Sorgente* e sostituendone i caratteri a partire dal carattere numero *Posizione*, con la stringa *Sostituzione*.

Attenzione: Non si tratta di un inserimento, ma di una sostituzione; i caratteri di *Sorgente* che si trovano a partire da *Posizione* vengono cancellati dalla stringa *Sostituzione* e non spostati. Non si tratta nemmeno della sostituzione di una stringa con un'altra. Sostituisci caratteri restituisce sempre una stringa di caratteri la cui lunghezza è uguale a quella di *Sorgente*.

Sostituisci caratteri restituisce una stringa non modificata nei seguenti casi:
- Se la stringa *Sorgente* è vuota.
- Se *Posizione* è minore o uguale a 0 o se *Posizione* è maggiore della lunghezza di *Sorgente*.

Esempio

```
Inizio:="ALLA VOSTRA SALUTE !"
DaInserire:="ALLEGRIA"
LaPosizione:=LaPosizione("SALUTE";Inizio)
Risultato:=Sostituisci caratteri(Inizio;DaInserire;LaPosizione)
```

dopo **Sostituisci caratteri**

vRisultato:=**Sostituisci caratteri**("Aci";"CI";2)	vRisultato assume il valore "ACI"
vRisultato:=**Sostituisci caratteri**("Novembre";"Dic";1)	vRisultato assume il valore "Dicembre"
vRisultato:=**Sostituisci caratteri**("ACI";" Domani";2)	vRisultato assume il valore "A D"

Riferimenti
Inserisci stringa, Cancella stringa, Sostituisci stringa

Cancella stringa — Delete string

Sintassi
Cancella stringa(Sorgente;Inizio;Lunghezza)
➤ Stringa

Parametri	Tipo	Descrizione
Sorgente	Espr_Alfa	Stringa di partenza
Inizio	Espr_NumPos	Posizione di partenza per la cancellazione
Lunghezza	Espr_Num	Numero dei caratteri da cancellare

Descrizione
Cancella stringa elimina un numero di caratteri uguale a *Lunghezza* a partire dalla posizione *Inizio*. Questa funzione non modifica la stringa *Sorgente*. Cancella stringa restituisce una stringa non modificata nei seguenti casi:
- Se la stringa *Sorgente* è vuota
- Se *Inizio* è minore o uguale a 0
- Se *Inizio* è maggiore o uguale alla lunghezza della stringa Sorgente
- Se *Lunghezza* è minore o uguale a 0.

Se *Lunghezza* è maggiore della lunghezza della stringa *Sorgente*, i caratteri verranno cancellati fino alla fine della stringa *Sorgente*.

Esempio

vSorgente:="ALLA VOSTRA BUONA SALUTE"
vInizio:=**Posizione**("BUONA";vSorgente)
vLunghezza:=**Lunghezza**("BUONA")+1
vRisultato:=**Cancella caratteri**(vSorgente;vInizio;vLunghezza)

vRisultato assumerà il valore:"ALLA VOSTRA SALUTE"

Riferimenti
Inserisci stringa, Sostituisci stringa

Inserisci stringa

Insert string

Sintassi
Inserisci stringa(Sorgente;DaInserire;Posizione)
➤ Stringa

Parametri	Tipo	Descrizione
Sorgente	Espr_Alfa	Stringa in cui si farà l'inserimento
DaInserire	Espr_Alfa	Stringa da inserire in Sorgente
Posizione	Espr_Num	Posizione dell'inserimento

Descrizione
Inserisci stringa restituisce una stringa di caratteri, introducendo *DaInserire* nella stringa *Sorgente* a partire da *Posizione*.
I caratteri dopo *Posizione* vengono quindi spostati alla fine della stringa inserita.

Inserisci stringa non esegue alcuna azione se:
- *DaInserire* è una stringa vuota
- *Posizione* è minore o uguale a zero.

Esempio

```
vSorgente:="ALLA VOSTRA SALUTE"
vDaInserire:="BUONA "
vPosizione:=Posizione("SALUTE";vSorgente)
vRisultato:=Inserisci stringa(vSorgente;vDaInserire;vPosizione)

vRisultato sarà uguale a:"ALLA VOSTRA BUONA SALUTE"
```

Riferimenti
Sostituisci stringa, Cancella stringa

Sostituisci stringa

Replace string

Sintassi
Sostituisci stringa(Sorgente; Precedente;Nuovo;{NumSost})
➤ Stringa

Parametri	Tipo	Descrizione
Sorgente	Espr_Alfa	Stringa in cui viene fatta la sostituzione
Precedente	Espr_Alfa	Stringa da sostituire in Sorgente
Nuovo	Espr_Alfa	Stringa che sostituirà Precedente
NumSost	Espr_NumPos	Numero delle sostituzioni da effettuare

Descrizione
Sostituisci stringa restituisce una stringa di caratteri che deriva dalla sostituzione, in *Sorgente*, di *Precedente* con *Nuovo*. Il numero di sostituzioni da effettuare può essere stabilito usando l'argomento opzionale *NumSost*. Se l'argomento *NumSost* viene omesso, *Sostituisci* stringa sostituirà ogni occorenza di *Precedente*.

Esempio

vSorgente:="ALLA VOSTRA SALUTE"
vRisultato:=**Sostituisci stringa**(vSorgente;"SALUTE";"BUONA SALUTE")

vRisultato assumerà il valore:"ALLA VOSTRA BUONA SALUTE"

Riferimenti
Inserisci stringa, Cancella stringa

Lunghezza — Lenght

Sintassi
Lunghezza(LaStringa)
➤ Intero

Parametri	Tipo	Descrizione
LaStringa	Espr_Alfa	Stringa di cui si vuole calcolare la lunghezza

Descrizione
Lunghezza restituisce il numero di caratteri di cui è composta *LaStringa*.

Esempio

a:="Buongiorno"
vLunghezza:=**Lunghezza**(a) — vLunghezza è uguale a 10

vLunghezza:=**Lunghezza**("Tremila lire") — vLunghezza è uguale a 12

Riferimenti
Posizione

Maiuscolo — Uppercase

Sintassi
Maiuscolo(LaStringa)
➤ Stringa

Parametri	Tipo	Descrizione
LaStringa	Espr_Alfa	Stringa da convertire in maiuscolo

Descrizione
Maiuscolo restituisce una stringa di caratteri uguale a *LaStringa*, convertita in maiuscolo.

Esempio

Maiuscolo("Buongiorno Signora") restituisce "BUONGIORNO SIGNORA"

Riferimenti
Minuscolo

Minuscolo — Lowercase

Sintassi
Minuscolo(LaStringa)
➤ Stringa

Parametri	Tipo	Descrizione
LaStringa	Espr_Alfa	Stringa da convertire in minuscolo

Descrizione
Minuscolo restituisce una stringa di caratteri uguale a *LaStringa*, convertita in minuscolo

Esempio

Minuscolo("ArRivedErci SiGnora") restituisce "arrivederci signora"

Riferimenti
Maiuscolo

Sotto_stringa — Substring

Sintassi
Sotto_stringa(LaStringa;Inizio;{Lunghezza})
➤ Stringa

Parametri	Tipo	Descrizione
LaStringa	Espr_Alfa	Stringa di partenza
Inizio	Espr_Num	Posizione di partenza
Lunghezza	Espr_Num	Numero di caratteri da estrarre

Descrizione
Sotto_stringa restituisce una stringa di caratteri composta dai caratteri di *LaStringa*, a partire dal carattere *Inizio* per un numero di caratteri pari a *Lunghezza*.
Se *Inizio* è maggiore della lunghezza di *LaStringa*, Sotto_stringa non esegue alcuna azione.

Se *Lunghezza* viene omessa, Sotto_stringa restituirà tutti i caratteri a partire dal carattere *Inizio* fino alla fine di *LaStringa*.

Esempio

Sotto_stringa("buon divertimento";6;5)	restituisce "diver"
Sotto_stringa("buon divertimento";20;5)	restituisce ""
Sotto_stringa("buon divertimento";6)	restituisce "divertimento"

Posizione — Position

Sintassi
Posizione(DaCercare;LaStringa)
➤ Intero

Parametri	**Tipo**	**Descrizione**
DaCercare	Espr_Alfa	Stringa da ricercare
LaStringa	Espr_Alfa	Stringa nella quale viene effettuata la ricerca

Descrizione
Posizione cerca la posizione della prima ricorrenza di *DaCercare* in *LaStringa*. Se *DaCercare* non viene trovato, Posizione restituisce il valore 0.

Esempio

Posizione("on";"buon divertimento")	restituisce 4
Posizione("Informatica";"buon divertimento")	restituisce 0
Posizione("n";"buon divertimento")	restituisce 5

Carattere — Char

Sintassi
Carattere(NumASCII)
➤ Stringa

Parametri	**Tipo**	**Descrizione**
NumASCII	Espr_Num	Numero del carattere ASCII desiderato

Descrizione
Carattere restituisce una stringa di caratteri composta da un solo carattere, il cui codice ASCII è uguale a *NumASCII*. Questa funzione vi serve principalmente per generare quei caratteri che non potete inserire da tastiera durante la programmazione, come ad esempio il tabulatore (9), il ritorno carrello (13), le virgolette (34).

Esempio

Carattere(65)	restituisce "A"
Carattere(13)	restituisce un carattere di ritorno carrello
Carattere(9)	restituisce un tabulatore
Carattere(34)	restituisce le virgolette

AVVISO("Buongiorno"+**Carattere**(13)+"E' il"+**Stringa**(**Data odierna**))

Riferimenti
Ascii

Ascii — Ascii

Sintassi
Ascii(LaStringa)
➤ Intero [0..255]

Parametri	**Tipo**	**Descrizione**
LaStringa	Espr_Alfa	Stringa di caratteri

Descrizione
Ascii restituisce il codice ASCII corrispondente al primo carattere di *LaStringa*.

Esempio

Ascii("A")	restituisce 65
Ascii("Anagramma")	restituisce 65

Riferimenti
Carattere

Stringa (applicata ai valori numerici) — String

Sintassi
Stringa(IlNumero;{LaFormattazione})
➤ Stringa

Parametri	**Tipo**	**Descrizione**
IlNumero	Espr_Num	Numero da convertire in stringa
LaFormattazione	Espr_Alfa	Formattazione

Descrizione
Stringa restituisce una stringa di caratteri composta dalle cifre di *IlNumero* più il separatore decimale, se è richiesto. *Formato* è opzionale, e permette di precisare la formattazione del valore numerico prima della conversione.

Esempio

Stringa(452,25)	restituisce "452,25"
Stringa(452,25;" ### ##0,00 FF")	restituisce "452,25 FF"
Stringa(-5;"Pos;Neg;Nul")	restituisce "Neg"

Riferimenti
Num

Stringa (applicata alle date) String

Sintassi
Stringa(LaData;{LaFormattazione})
➤ Stringa

Parametri	**Tipo**	**Descrizione**
LaData	Espr_Data	Data da convertire in stringa
LaFormattazione	Espr_Num	Formattazione della data

Descrizione
Stringa restituisce una stringa di caratteri uguale a *LaData* sotto forma alfanumerica.
La formattazione assume i valori 1,2,3 o 4 in base ai 4 formati predeterminati disponibili.

Esempio

Stringa(Data odierna)	restituisce "14/07/89", il 14 luglio 1989

Riferimenti
Data

Osservazioni relative ai booleani

La funzione **Stringa** non può essere usata per effettuare la conversione dei booleani in stringhe di caratteri. Se vi occorre questa funzione di conversione in "0" o "1", dove $1 è il booleano da trasformare, scriverete:

$0:=**Stringa**(**Num**($1))

Se volete trasformare il booleano in unastringa di caratteri uguale a "Vero" o "Falso", dove $1 è il booleano da convertire, scriverete:

$0:=(**Num**($1=**Vero**)*"Vero")+(**Num**($1=**Falso**)*"Falso")

Se volete passare voi stessi i due valori di ritorno:

$0:=(**Num**($1=**Vero**)*$2)+(**Num**($1=**Falso**)*$3)

Esempio
Tit:=*BoolInStringa3* (x=1;"Maschio";"Femmina").

Date e ore

Data odierna

Current date

Sintassi
Data odierna
➤ Data

Descrizione
Data odierna restituisce una data corrispondente a quella impostata nel Pannello di Controllo del vostro Macintosh. Controllate quindi che il datario sia regolato in modo esatto, se volete che questa funzione vi ritorni una data utilizzabile.

Esempio

Data odierna	restituisce !14/7/89! il 14 luglio 1989 e non 1789

Se(**Data odierna** =!14/07/89!)	Se è il 14 luglio 1989
SUONA("Inno di Mameli")	Suona l'inno nazionale italiano
Fine se	

Riferimenti
Data

Stringa in data

Date

Sintassi
Stringa in data(LaStringa)
➤ Data

Parametri	Tipo	Descrizione
LaStringa	Espr_Alfa	Stringa di caratteri da convertire in data

Descrizione
Stringa in data restitusce una data ottenuta effettuando la conversione di LaStringa in data

Esempio

Stringa in data("3/8/89") .	restituisce !03/08/89!

$a:=**Richiedi**("Data di partenza ?")	Richiedi la data di partenza
Se(OK=1)	Se la data è confermata
Ladata:=**Stringa in data**($a)	Converti $a in data
Fine se	

Immaginiamo ora una procedura generale di richiesta di una data, *RichiediData*.

```
$Risposta:=Richiedi($1;Stringa($2))          $1 corrisponde al messaggio, $2 alla data corrente
Se(OK=1)                                     Se la richiesta è confermata
    $0:=Stringa in data($Risposta)           Converti $0 in data
altrimenti
    $0:=!00/00/00!                           Altrimenti restituisci una data nulla
Fine se
```

Potremo in seguito utilizzare questa procedura, scrivendo
LaData:=RichiediData("Data di partenza?";**Data odierna**)

Riferimenti
Stringa, Data odierna

Anno di — Year of

Sintassi
Anno di(LaData)
➤ Numerico

Parametri	Tipo	Descrizione
LaData	Espr_Data	Data della quale volete conoscere l'anno

Descrizione
Anno di restituisce un valore numerico corrispondente all'anno di *LaData*

Esempio

Anno di(!15/05/89!) restituisce 1989

Immaginiamo una funzione che aggiunga un anno ad una data passata come parametro

$0:=**Stringa in data**(**Sotto_stringa**(**Stringa**($1);0;5)+**Stringa**(Anno di($1)+1))

AnnoProssimo:=*AggiungiAnno*(**Data odierna**) Chiamata di *AggiungiAnno*

Riferimenti
Mese di, Giorno di, Giorno della settimana

Mese di — Month of

Sintassi
Mese di(LaData)
➤ Numerico

Parametri	Tipo	Descrizione
LaData	Espr_Data	Data di cui si vuole conoscere il mese

Descrizione
Mese di restituisce un valore numerico corrispondente al mese di *LaData*.

Esempio

Mese di(!15/05/89!)	restituisce 5
$0:=*Data*("1/"+**Stringa**(**Mese di**($1))+"/"+**Stringa**(**Anno di**($1)))	La funzione *PrimoDelMese* restituisce una data corrispondente al primo giorno del mese
LaData:=PrimoDelMese(**Data odierna**)	Chiamata di *PrimoDelMese*

Riferimenti
Anno di, Giorno del mese, Giorno della settimana

Giorno del mese — Day of

Sintassi
Giorno del mese(LaData)
➤ Numerico

Parametri	Tipo	Descrizione
LaData	Espr_Data	Data di cui si vuole conoscere il giorno del mese

Descrizione
Giorno del mese restituisce un valore numerico corrispondente al giorno del mese di *LaData*

Esempio
Giorno del mese(!15/05/89!) restitusice 15

Riferimenti
Mese di, Anno di, Giorno della settimana

Giorno della settimana — Day number

Sintassi
Giorno della settimana(LaData)
➤ Numerico

Parametri	Tipo	Descrizione
LaData	Espr_Data	Data di cui si vuole conoscere il numero del giorno nella settimana

Descrizione
Giorno della settimana restituisce un valore numerico pari al giorno della settimana di *LaData*.
La numerazione dei giorni della settimana viene effettuata nel modo seguente:

Domenica	➤ 1
Lunedì	➤ 2
Martedì	➤ 3
Mercoledì	➤ 4
Giovedì	➤ 5
Venerdì	➤ 6
Sabato	➤ 7

Esempio
La procedura seguente crea un vettore di stringhe che chiameremo Giorno, contenente il nome dei giorni della settimana.

```
VETTORE DI STRINGHE(10;Giorno;7)
Giorno{1}:="Domenica"
Giorno{2}:="Lunedì"
Giorno{3}:="Martedì"
Giorno{4}:="Mercoledì"
Giorno{5}:="Giovedì"
Giorno{6}:="Venerdì"
Giorno{7}:="Sabato"
```

La procedura *NuovoGiorno* esegue l'assegnamento del vettore Giorno.

E' sufficiente chiamare la procedura *NuovoGiorno* una volta sola, quando avviate la base dati, per poter utilizzare il vettore corrispondente in seguito.

```
AVVISO(Giorno{(Giorno della settimana(Data odierna))})
```

Chiamata della funzione AVVISO contenente il nome del giorno della data corrente

La funzione seguente, *InizioSettimana*, restituisce una data uguale al primo giorno della settimana della data passata come argomento

```
$0:=$1-Giorno della settimana($1)+1
```

Funzione *InizioSettimana*

```
IlLunedì:=InizioSettimana(Data odierna)
```

Esempio di chiamata della funzione InizioSettimana

Riferimenti
Anno di, Mese di, Giorno del mese

Ora attuale — Current time

Sintassi
Ora attuale
➤ Tempo

Descrizione
Ora attuale è una funzione che restituisce un valore temporale

Riferimenti
Ora in stringa

Ora in stringa — Time string

Sintassi
Ora in stringa(Ora)
➤ Stringa

Parametri	Tipo	Descrizione
Ora	Espr_Ora	Ora da trasformare in stringa

Descrizione
Ora in stringa restituisce un valore alfanumerico che esprime l'ora passata come argomento da *Ora*.

Riferimenti
Ora attuale, Stringa in ora

Stringa in ora — Time

Sintassi
Stringa in ora(Stringa)
➤ Ora

Parametri	Tipo	Descrizione
Stringa	Espr_Alfa	Stringa da convertire in ora

Descrizione
Stringa in ora restituisce un valore temporale che esprime l'ora passata come argomento da *Stringa*.

Riferimenti
Ora attuale, Ora in stringa

Booleani

Non ## Not

Sintassi
Non(lIBool)
➤ Booleano

Parametri	Tipo	Descrizione
lIBool	Espr_Bool	Espressione booleana da negare

Descrizione
Non restituisce la negazione dell'espressione booleana *lIBool.*

Esempio

```
Non(2=2)                                  restituisce Falso
Non(Vero)                                 restituisce Falso

IMPOSTA ARCHIVIO CORRENTE([Clienti])
TUTTE LE SCHEDE                           Seleziona tutte le schede
Mentre(Non(Fine selezione))               Mentre Fine selezione è falso
   MOSTRA SCHEDA                          Visualizza la scheda corrente
   SCHEDA SUCCESSIVA                      Passa alla scheda successiva
Fine mentre                               Fine del ciclo di Mentre
```

Vero ## True

Sintassi
Vero
➤ Booleano

Descrizione
Vero restituisce un valore booleano uguale a vero.

Falso ## False

Sintassi
Falso
➤ Booleano

Descrizione
Falso restituisce un valore booleano uguale a falso.

Funzioni matematiche

Ass — Abs

Sintassi
Ass(ValNumerico)
➤ Numerico

Parametri	Tipo	Descrizione
ValNumerico	Espr_Num	Valore da convertire

Descrizione
Ass restituisce il valore assoluto di *ValNumerico*. In altri termini, se *ValNumerico* è negativo, Ass lo restituirà positivo e se *ValNumerico* è positivo, non verrà modificato.

Esempio

Ass(12,35)	restituisce 12,35
Ass(-5)	restituisce 5

Riferimenti
Int, Dec, Arrotonda

Int — Int

Sintassi
Int(ValNumerico)
➤ Numerico

Parametri	Tipo	Descrizione
ValNumerico	Espr_Num	Valore da convertire

Descrizione
Int restituisce la parte intera di *ValNumerico* senza arrotondamenti.

Esempio

Int(15,742)	restituisce 15
Int(-325,46)	restituisce -326

Riferimenti
Dec, Int, Arrotonda

Dec

Dec

Sintassi
Dec(ValNumerico)
➤ Numerico

Parametri	Tipo	Descrizione
ValNumerico	Espr_Num	Valore da convertire

Descrizione
Dec restituisce la parte decimale di *ValNumerico*. Il valore restituito sarà sempre positivo o nullo.

Esempio

Dec(156,12)	restituisce 0,12
Dec(-5,8456)	restituisce 0,8456

Riferimenti
Int, Arrotonda, Ass

Num (applicato alle stringhe di caratteri)

Num

Sintassi
Num(LaStringa)
➤ Numerico

Parametri	Tipo	Descrizione
LaStringa	Espr_Alfa	Stringa da convertire in numero

Descrizione
Num restituisce un valore numerico ottenuto convertendo *LaStringa*. Se *LaStringa* non rappresenta un numerico, Num restituirà 0. Quando utilizzate la funzione Num avete a disposizione tre caratteri riservati:
- "e", considerato come esponente di un valore numerico rappresentato in notazione scientifica,
- il trattino "-" interpretato come il segno meno,
- la virgola "," considerata come separatore decimale.

Esempio

Num("1542,25")	restituisce 1542,25
Num("Tagada")	restituisce 0
Num("Toto025")	restituisce 25

Attenzione: Non usate questa funzione per estrarre le cifre contenute in una stringa di caratteri, perchè se questa contiene uno dei 3 caratteri riservati, l'estrazione non sarà corretta.

Riferimenti
Stringa

Num (applicato ai booleani) — Num

Sintassi
Num(lIBool)
➤ Intero [0,1]

Parametri	Tipo	Descrizione
lIBool	Espr_Bool	Booleano da convertire in numero

Descrizione
Num applicato a *lIBool* restituisce un valore numerico uguale a 1 se *lIBool* è vero, e uguale a 0 se *lIBool* è falso.

Esempio

Num(Vero)	retourne 1
Num(Falso)	retourne 0
Num(Voto>10)*"Bene"+**Num**(Voto<10)*"Male"	restituisce "Bene" se Voto è maggiore di 10, restituisce "Male" se Voto è minore di 10.

Arrotonda — Round

Sintassi
Arrotonda(ValNumerico;NumDecimali)
➤ Numerico

Parametri	Tipo	Descrizione
ValNumerico	Espr_Num	Valore da arrotondare
NumDecimali	Espr_Num	Numero di decimali dell'arrotondamento

Descrizione
Arrotonda restituisce il valore arrotondato di *ValNumerico*, con una precisione uguale a *NumDecimali*.
Se *NumDecimali* è positivo, l'arrotondamento viene effettuato sulla parte decimale di *ValNumerico*
Se *NumDecimali* è negativo, l'arrotondamento viene effettuato sulla parte intera di *ValNumerico*

ValNumerico	#	#	#	#	#	#	#
NumDecimali	-3	-2	-1	0	1	2	3

Esempio

Arrotonda(7,1543986;2)	restituisce 7,15
Arrotonda(- 7,1543986;2)	restituisce -7,16
Arrotonda(32345,67;-3)	restituisce 32000

Riferimenti
Int, Dec, Ass

Tronca — Trunc

Sintassi
Tronca(ValNumerico;NumDecimali)
➤ Numerico

Parametri	Tipo	Descrizione
ValNumerico	Espr_Num	Numero da troncare
NrDecimali	Espr_Num	Numero di decimali

Descrizione
Tronca restituisce un valore numerico corrispondente a *ValNumerico*, troncato a partire dal decimale *NumDecimali*.Se *NumDecimali* è positivo, il troncamento avviene sulla parte decimale di *ValNumerico*. Se *NumDecimali* è negativo, il troncamento avviene sulla parte intera di *ValNumerico*.

Esempio

Tronca(5,45701;2)	restitusce 5,45
Tronca(-7,1543986;3)	restituisce -7,154
Tronca(-216,887;-1)	restituisce -220

Riferimenti
Arrotonda, Int, Dec

Modulo — Mod

Sintassi
Modulo(IlValore;Divisore)
➤ Numerico

Parametri	Tipo	Descrizione
IlValore	Espr_NumPos	Valore di cui si deve calcolare il modulo
Divisore	Espr_NumPos	Divisore

Descrizione
Modulo restituisce il resto della divisione intera di il *Valore* per *Divisore*.

Esempio

Modulo(8;2)	restituisce 0
Modulo(10;3)	restituisce 1

```
Ciclo(i;1;1000)                    Esegui un ciclo da 1 a 1000
  Se(Modulo(i;50)=0)               Se Modulo di i su 50 è uguale a 0 (cioé se i è un multiplo di 50)
     Messaggio(Stringa(i))         Visualizza il valore di i
  Fine se
Fine ciclo
```

Casuale — Random

Sintassi

Casuale

➤ Intero [0..32767]

Descrizione

Casuale restituisce un numero intero aleatorio, compreso tra 0 e 32767 (limiti inclusi). Per ottenere un numero compreso tra due valori di vostra scelta, seguite l'esempio indicato.

Esempio

Questo esempio riguarda la procedura *CasualeFra*, che restituisce un valore compreso tra due limiti da voi stabiliti.

```
$0:=Modulo(Casuale;($2-$1+1)) + $1      $1 corrisponde al limite massimo
                                        $2 corrisponde al limite minimo
```

Potrete in seguito chiamare la procedura *CasualeFra* per ottenere un valore compreso, ad esempio, tra 10 e 100

v:=*CasualeFra*(10;100)

Cos — Cos

Sintassi

Cos(Angolo)

➤ Numerico

Parametri	**Tipo**	**Descrizione**
Angolo	Espr_Num	Angolo espresso in radianti

Descrizione

Cos restituisce il coseno di *Angolo* espresso in radianti

Riferimenti

Sen, Tan, Arctan

Sen

Sin

Sintassi
Sen(Angolo)
➤ Numerico

Parametri	Tipo	Descrizione
Angolo	Espr_Num	Angolo espresso in radianti

Descrizione
Sen restituisce il seno di *Angolo* espresso in radianti.

Riferimenti
Cos, Tan, Arctan

Tan

Tan

Sintassi
Tan(Angolo)
➤ Numerico

Parametri	Tipo	Descrizione
Angolo	Espr_Num	Angolo espresso in radianti

Descrizione
Tan restituisce la tangente di *Angolo* espresso in radianti

Riferimenti
Cos, Sen, Arctan

Arctan

Arctan

Sintassi
Arctan(ValNumerico)
➤ Angolo

Parametri	Tipo	Descrizione
ValNumerico	Espr_Num	Numero la cui tangente è uguale ad Angolo

Descrizione
Arctan restituisce l'arcotangente di *ValNumerico* espressa in radianti.

Riferimenti
Sen, Tan

Log — Log

Sintassi
Log(Numero)
➤ Numerico

Parametri	Tipo	Descrizione
Numero	Espr_Num	Numero positivo non nullo

Descrizione
Log restituisce il logaritmo naturale (in base e) di *Numero*.

Esempio

Log(1)	restituisce 0
Log(10000)/**Log**(10)	restituisce 4

Riferimenti
Esp

Esp — Exp

Sintassi
Esp(Numero)
➤ Numerico

Parametri	Tipo	Descrizione
Numero	Espr_Num	Numero positivo non nullo

Descrizione
Esp eleva la base dei logaritmi naturali (e) alla potenza di *Numero*.

Esempio

Esp(1)	restituisce 2,718281828459045

Riferimenti
Log

Vettori

Introduzione

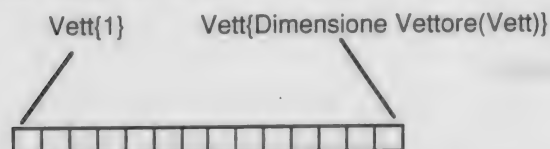
Se affrontate l'argomento dei vettori per la prima volta, consultate la sezione relativa ai vettori nel manuale Ambiente Progetto.
Abbiamo preferito far precedere l'analisi dei comandi di gestione dei vettori da una panoramica generale sulle variabili di tipo vettore ed una descrizione dell'uso dei vari comandi.

Caratteristiche generali dei vettori

Le variabili di tipo vettore sono variabili che contengono un numero qualsiasi (limitato dalle dimensioni della memoria) di elementi dello stesso tipo. In effetti, potete avere dei vettori di reali, di interi, di interi lunghi, di booleani, di testi, di stringhe, di immagini, di date o di puntatori. Queste variabili devono essere dichiarate utilizzando rispettivamente i comandi VETTORE DI REALI, VETTORE DI INTERI, VETTORE DI INTERI LUNGHI, VETTORE DI BOOLEANI, VETTORE DI TESTI, VETTORE DI IMMAGINI, VETTORE DI STRINGHE, VETTORE DI DATE o VETTORE DI PUNTATORI.
Quando dichiarate un vettore, ne stabilite la dimensione di partenza.
Esempio di dichiarazione di un vettore con 10 elementi:

VETTORE DI TESTI(MioVettore;10)

Questa dimensione può essere modificata in qualsiasi momento
- sia direttamente: richiedete una nuova dimensione con i comandi di dimensionamento
- sia indirettamente: utilizzate dei comandi che modifichino il numero di elementi, come ad esempio INSERISCI ELEMENTI IN VETTORE, CANCELLA ELEMENTI VETTORE, SELEZIONE IN VETTORE, COPIA VETTORE, LISTA SCELTE IN VETTORE.

Gli elementi di un vettore hanno come riferimento il proprio numero d'ordine, compreso tra 1 e la dimensione del vettore, in questo caso tra 1 e 10. Per accedere ad un elemento, scrivete il nome del vettore seguito dal numero d'ordine dell'elemento tra parentesi graffe, ad esempio Vett{3}:
Le parentesi graffe si ottengono premendo **Opzione-(** e **Opzione-)**.

Vett{Dimensione Vettore(Vett)} corrisponde all'accesso dell'ultimo elemento del vettore.

Un vettore viene creato in memoria. Se riducete la dimensione di un vettore mentre lo state usando, gli elementi che non sono compresi nelle nuove dimensioni vengono cancellati dalla memoria.

Occupazione di memoria di un vettore:

VETTORE DI TESTI	8+(NrElementi*4)+Somma delle lunghezze dei testi registrati nel vettore
VETTORE DI STRINGHE	8+(NrElementi*lunghezza di definizione di un elemento)
VETTORE DI REALI	8+(NrElementi*10)
VETTORE DI INTERI	8+(NrElementi*2)
VETTORE DI INTERI LUNGHI	8+(NrElementi*4)
VETTORE DI BOOLEANI	8+(NrElementi/8)
VETTORE DI DATE	8+(NrElementi*6)
VETTORE DI IMMAGINI	8+(NrElementi*4)+Somma delle dimensioni delle immagini registrate nel vettore
VETTORE DI PUNTATORI	8+(NrElementi*16)

Osservazioni sull'uso delle variabili di tipo vettore

- Quando utilizzate la struttura di controllo Ciclo, il contatore deve essere una variabile di tipo numerico e non può in nessun caso essere una variabile di tipo vettore.
- Gli operatori di confronto vi permettono di paragonare gli oggetti gestiti in 4th DIMENSION e non possono essere applicati ai vettori ma soltanto ai loro componenti.
- Potete passare ad una procedura variabili e campi di tutti i tipi, ad eccezione delle variabili di tipo vettore nella loro forma completa. Potete invece passare un elemento dellla variabile di tipo vettore.
- Se passate il nome di un vettore come argomento ad una procedura, non viene passato il nome del vettore, bensì l'elemento corrente nel vettore stesso. Tale valore è significativo soltanto se il vettore appare sullo schermo sotto forma di area di scorrimento o di menu a comparsa.
- Non esistono variabili locali di tipo vettore.
- Una funzione non può ritornare vettori. Se volete ottenere un vettore, usate le variabili di tipo puntatore, descritte più avanti in questo stesso manuale.

Gestione di un vettore

Il linguaggio di 4th DIMENSION vi fornisce le funzioni standard per la gestione dei vettori, che vi permettono di aggiungere, cancellare, inserire, ordinare e cercare valori.
Per semplificare l'operazione di riempimento dei vettori, avete a disposizione due coppie di funzioni, con le quali potete:
a/ trasferire informazioni da un vettore ad una lista scelte e viceversa;
b/ trasferire informazioni da un vettore ad una selezione di schede, e viceversa;

Per aggiungere un valore alla fine di un vettore

L'aggiunta di un valore all'interno di un vettore si ottiene aumentando le dimensioni del vettore utilizzato. A questo scopo, avete a disposizione diversi metodi:

1) l'inserimento di una riga a fine vettore

INSERISCI ELEMENTI IN VETTORE(MioVettore;**Dimensione Vettore**(MioVettore)+1)
MioVettore{**Dimensione Vettore**(MioVettore)}:=MioValore

2) il cambiamento della dimensione del vettore in base alla Dimensione Vettore+1

VETTORE DI TESTI(MioVettore;**Dimensione Vettore**(MioVettore)+1)
MioVettore{**Dimensione Vettore**(MioVettore)}:=MioValore

3) chiamando una procedura che utilizza un parametro di tipo puntatore.

Questa procedura, che possiamo chiamare MiaAggiunta, si scrive nel modo seguente:
INSERISCI ELEMENTI IN VETTORE($1»;**Dimensione Vettore**($1»)+1)
$1»{**Dimensione Vettore**($1»)}:=$2

dove $1 è un puntatore al vettore e $2 il valore da aggiungere.

L'aggiunta si ottiene in questo modo:
MiaAggiunta(»MioVettore;MioValore)

Per cancellare un elemento del vettore

La cancellazione di un elemento del vettore si ottiene utilizzando la procedura CANCELLA ELEMENTI VETTORE, e passando come argomento il nome del vettore e, successivamente, il numero dell'elemento da cancellare:
CANCELLA ELEMENTI VETTORE(MioVettore;NumElemento;NumCanc)

NumCanc è il numero di elementi da cancellare. Si tratta di un argomento opzionale. Se lo omettete, verrà cancellata una sola riga.

Per cancellare l'elemento 3 del vettore, scriverete:
CANCELLA ELEMENTI VETTORE(MioVettore;3)

Per cancellare l'ultima riga del vettore, scriverete:
CANCELLA ELEMENTI VETTORE(MioVettore;**Dimensione Vettore**(MioVettore))

Per inserire righe in un vettore

L'inserimento di righe in un vettore si ottiene semplicemente chiamando il comando INSERISCI ELEMENTI IN VETTORE.
Dovrete precisare in quale punto volete ottenere l'inserimento, e quante righe volete inserire.
INSERISCI ELEMENTI IN VETTORE(MioVettore;NumElemento;NumIns)

NumIns è il numero di elementi da inserire. Si tratta di un argomento opzionale. Se lo omettete, verrà inserita una sola riga.

Inserite una riga in prima posizione nel vettore *MioVettore*:
INSERISCI ELEMENTI IN VETTORE(MioVettore;1)

Inserite 2 righe a partire dalla posizione 3 del vettore MioVettore:
INSERISCI ELEMENTI IN VETTORE(MioVettore;3;2)

Per ordinare un vettore

L'ordinamento di un vettore si esegue con il comando ORDINA VETTORE.
Si tratta di un ordinamento molto rapido, poichè viene effettuato interamente nella memoria centrale.
Volete ordinare un vettore in ordine crescente:
ORDINA VETTORE(MioVettore;>)

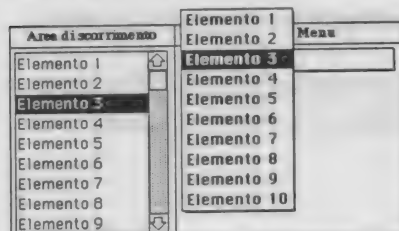
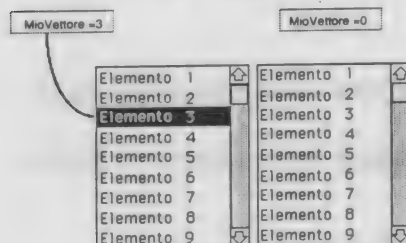
Per eseguire una ricerca all'interno di un vettore

La ricerca in un vettore si esegue con la funzione **Cerca in vettore**.
Passate come argomento il nome del vettore nel quale volete fare la ricerca, poi il valore da cercare.
Cerca in vettore restituisce il numero del primo elemento che corrisponde alla vostra ricerca.
Se non trova nessun elemento, **Cerca in vettore** restituirà il valore -1.

Esempio di ricerca di un valore in un vettore di testi:
Trovato:=**Cerca in vettore**(MioVettore;"Buongiorno")

Una ricerca all'interno di un vettore non crea una selezione come l'equivalente ricerca all'interno di un archivio.

Rappresentazioni dei vettori in un formato

Un vettore può comparire in un formato sotto due forme diverse: come un'area di scorrimento, o come menu a comparsa. In entrambi i casi, è sufficiente creare una variabile, assegnarle il nome del vettore e, come tipo, l'area di scorrimento o il menu a comparsa.

Quando il vettore IlVettore appare sullo schermo, la variabile IlVettore contiene il numero dell'elemento correntemente selezionato.

Vettori sincronizzati

4th DIMENSION vi permette di far comparire diverse aree di scorrimento raggruppate fra loro.
Raggruppate significa che lo scorrimento sarà contemporaneo per tutti questi vettori e che alla destra dell'area raggruppata comparirà una sola barra di scorrimento.
A questo scopo, è sufficiente creare diverse aree di scorrimento, come indicato in questo schema:

Dopo aver creato queste 3 aree con le relative istruzioni, selezionatele e scegliete **Raggruppa** nel menu **Oggetti**.
Quando il formato comparirà sullo schermo, i vettori si presenteranno in questo modo:

Aree di scorrimento raggruppate

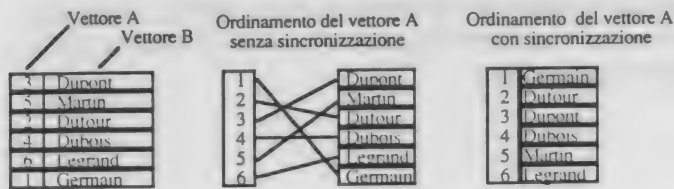
Dupont	25	Tecnico
Dufour	35	Ragioniere
Legrand	28	Dir Tecnico
Dupuis	22	Commerciale
De la tour	26	Operaio
Manieux	35	Magazziniere

Durante la selezione di una riga, 4th DIMENSION selezionerà automaticamente le righe corrispondenti negli altri vettori.
Se desiderate modificare l'ordine di visualizzazione di un vettore, usate il comando ORDINA VETTORE.
Se ordinate il vettore "a" servendovi del comando ORDINA VETTORE(a;>), il vostro formato avrà l'aspetto seguente:

Aree di scorrimento raggruppate

De la tour	25	Tecnico
Dufour	35	Ragioniere
Dupont	28	Dir Tecnico
Dupuis	22	Commerciale
Legrand	26	Operaio
Manieux	35	Magazziniere

Il vettore "a" è stato ordinato correttamente, ma i vettori "b" e "c" hanno conservato il loro ordine precedente.
Per risolvere questo problema, il comando ORDINA VETTORE può accettare come argomento diversi vettori.

Il primo di essi sarà ordinato, mentre gli altri vettori saranno sincronizzati con il primo.

Potete scrivere l'ordinamento sincronizzato passando al comando ORDINA VETTORE la lista dei vettori da ordinare, seguita dalla sequenza di ordinamento desiderata (>crescente, <decrescente). Il primo vettore della lista sarà effettivamente ordinato, gli altri verranno riorganizzati in base all'ordinamento del primo vettore.
Ordiniamo i vettori a,b,c, partendo dai valori del vettore a:
ORDINA VETTORE(a;b;c;>)

Vettore a due dimensioni

Caso generale

4th DIMENSION permette la dichiarazione di vettori a due dimensioni. Passate quindi 2 parametri di dimensionamento al comando.
Esempio
VETTORE DI TESTI(NomeVettore;Par1;Par2)

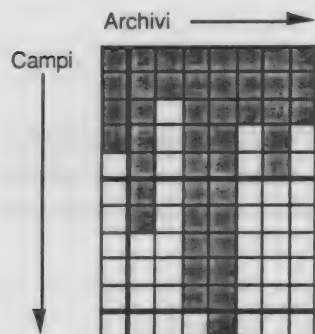
L'accesso ad un elemento del vettore presuppone che indichiate il numero della colonna tra parentesi graffe, seguito dal numero della riga, ugualmente tra parentesi graffe.
La colonna rappresenta la prima dimensione, e la riga la seconda.
Un vettore a due dimensioni non può essere raffigurato graficamente in un formato, perciò dovete creare due vettori ad una dimensione.
Se si crea uno schema di ciò che succede, si otterrà una rappresentazione di questo tipo:

Colonne

Righe	Francia	Germania	Inghilterra	XXXX
	Bonjour	Guten Tag	Hello	

Se volete accedere alla riga 3 della colonna 1, scrivete:
MioVettore{1}{3}
Il principio di creazione dei vettori a due dimensioni è molto semplice. Tuttavia, poichè i vettori sono gestiti in memoria centrale, può essere utile cercare di ottimizzare la gestione del dimensionamento del vettore.

Esempio di ottimizzazione

Immaginiamo di dichiarare un vettore a due dimensioni chiamato *Strutt* per rappresentare la struttura della base dati, cioé gli archivi ed i campi. Potreste predeterminare le dimensioni del vettore, assegnando il numero degli archivi e prendendo come riferimento il numero massimo di campi all'interno dell'archivio. Uno schema darebbe una rappresentazione di questo tipo:

Il vettore presenta elementi inutili corrispondenti a quello che sarebbe lo spazio occupato dai campi se tutti gli archivi avessero lo stesso numero di campi. Per risparmiare memoria, potete procedere ad una gestione del dimensionamento adatta ad ogni singolo archivio.
Cominciate a dichiarare la vostra prima dimensione.
Una funzione di 4th DIMENSION restituisce il numero degli archivi: **Numero di archivi.**
Nella seconda dimensione, si considera automaticamente l'esistenza di un solo campo.

Scriverete, quindi:
VETTORE DI TESTI(Strutt;**Numero di archivi**;1)

Creiamo poi un Ciclo che ci restituisca il numero degli archivi nella base dati tenendo conto del loro numero.
Ciclo(NumArch;1;**Numero di archivi**)

Potete dichiarare la seconda dimensione, il numero dei campi, che viene restituito dalla funzione di 4th DIMENSION **Numero di campi**.
I parametri passati al comando VETTORE DI TESTI saranno i seguenti:
(Strutt{NumArch};**Numero di campi**(NumArch))
La vostra procedura si presenterà in questo modo:

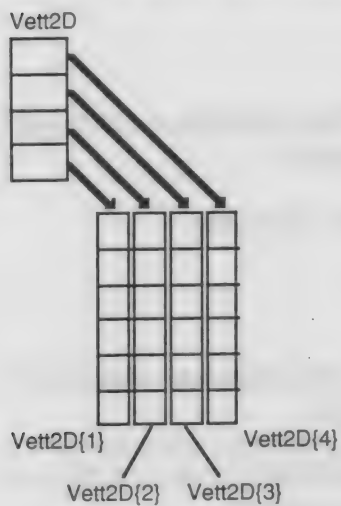
```
VETTORE DI TESTI(Strutt;Numero di archivi;1)
Ciclo(NumArch;1;Numero di archivi)
   VETTORE DI TESTI(Strutt{NumArch};Numero di campi(NumArch))
Fine ciclo
```

Nella dichiarazione dei vettori, vengono ormai presi in considerazione soltanto gli elementi "utili".

Nota
La seguente procedura vi permette di eseguire una dichiarazione ottimizzata del vettore.
Se volete anche assegnare i valori, è sufficiente assegnarli al vettore.
4th DIMENSION verifica i campi di ogni archivio in un ciclo situato all'interno del ciclo iniziale.

```
VETTORE DI TESTI(Strutt;Numero di archivi;1)
Ciclo(NumArch;1;Numero di archivi)
    VETTORE DI TESTI(Strutt{NumArch};Numero di campi(NumArch))
    Ciclo(NumCampi;1;Numero di campi(NumArch))
        Strutt{NumArch}{NumCampi}:=Nome del campo(NumArch;NumCampo)
    Fine ciclo
Fine ciclo
```

Gestione standard di un vettore a due dimensioni

Le gestioni di base su un vettore a due dimensioni non sono diverse da quelle su un vettore ad una dimensione. Partiamo da uno schema che illustra la rappresentazione interna di un vettore a due dimensioni.
Si tratta, in effetti, di un vettore di vettori.

Quando applicate un comando di gestione vettori ad un vettore a due dimensioni, questo si riferirà al vettore principale.

Perciò, il comando:
INSERISCI ELEMENTI IN VETTORE(Vett2D;2)
non inserirà un elemento in ogni colonna prima del secondo elemento, ma inserirà una colonna prima di Vett2D{2}.
Se volete inserire un elemento prima dell'elemento 2 nella colonna 2, dovrete scrivere:
INSERISCI ELEMENTI IN VETTORE(Vett2D{2};2)
Se invece desiderate inserire un elemento che si trovi in ogni colonna in posizione due, scriverete:

```
Ciclo(Colonna;1;Dimensione vettore(Vett2D))
   INSERISCI ELEMENTI IN VETTORE(Vett2D{Colonna};2)
Fine ciclo
```

Questo tipo di gestione è valido per tutte le operazioni di modifica di un vettore: CANCELLA ELEMENTI VETTORE, ORDINA VETTORE.

Vettore a due dimensioni in un formato

Un vettore a due dimensioni non può comparire come tale in un formato.
E' comunque possibile ottenerlo.
Se volete far apparire una o più colonne di un vettore a due dimensioni, usate il comando **COPIA VETTORE** e ricopiate ogni colonna in un vettore ad una dimensione.
Se il vettore a due dimensioni contiene, ad esempio, 3 colonne e volete farlo comparire sullo schermo, create una procedura *AggiornaVett*, che effettuerà il trasferimento.

```
COPIA VETTORE(Vett2D{1};Vett1)
COPIA VETTORE(Vett2D{2};Vett2)
COPIA VETTORE(Vett2D{3};Vett3)
```

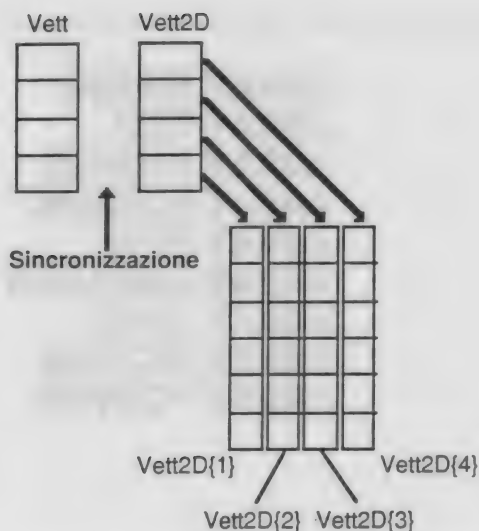
Ad ogni modifica di un elemento del vettore a due dimensioni, chiamate la procedura *AggiornaVett* per ottenere una corretta visualizzazione sullo schermo.

Ordinamento di un vettore a due dimensioni

L'ordinamento di un vettore a due dimensioni nella forma **ORDINA VETTORE**(Vett2D;>) non avrebbe senso, in quanto non potete sapere su quali valori eseguire tale ordinamento.
Potete invece ordinare una colonna di un vettore a due dimensioni.
Immaginate di dover ordinare la prima colonna di un vettore a due dimensioni. Scriverete:

ORDINA VETTORE(Vett2D{1};>)

Ordinamento sincronizzato di un vettore a due dimensioni

Potete ordinare un vettore e richiedere la sincronizzazione di un vettore a due dimensioni. Potete ordinare un vettore e richiedere la sincronizzazione di un vettore a due dimensioni.
La sincronizzazione non modificherà la sequenza dei valori all'interno di ogni colonna, ma scambierà le colonne all'interno del vettore a due dimensioni. Scrivendo **ORDINA VETTORE**(Vett;Vett2D;>), la sincronizzazione avviene tra Vett e Vett2D, e non per i valori di ogni singola colonna.

Questa sequenza di ordinamento darà luogo ad uno scambio di colonne, senza modificare la sequenza dei valori di Vett2D{1},...,Vett2D{4}.

Se volete ottenere una sincronizzazione dei valori contenuti nel vettore, dovete scriverla in questa forma:
ORDINA VETTORE(Vett;Vett2D{1};Vett2D{2};Vett2D{3};Vett2D{4};>)

Gestione del doppio clic in un'area di scorrimento

4th DIMENSION non è dotato di comandi che permettano di intercettare direttamente un doppio clic in un'area di scorrimento.
Ecco uno script che potrete introdurre nella vostra area di scorrimento, e che simulerà la gestione di un doppio clic. Si basa sull'intervallo tra due clic all'interno dell'area, e dipende quindi dal Macintosh che usate.
In questo esempio, l'intervallo impostato è di 2 secondi. Se fosse troppo breve, aumentate il valore 2 nel confronto:
((Ora attuale-TempoPrec)<2)

```
Se(Prima)                                        Se nel ciclo di Prima
    NumPrec:=0                                   Inizializza le due variabili
    TempoPrec:=†00:00:00†
Fine se
Se(Durante)                                      Se nel ciclo di Durante
    Se(NumPrec=MioVettore)&(Ora Attuale-TempoPrec<2)   Se l'intervallo è inferiore a due secondi e
    ...                                          l'elemento non cambia è un doppio clic
    Fine se
Fine se
NumPrec:=MioVettore
TempoPrec:=Ora Attuale
```

VETTORE DI STRINGHE — ARRAY STRING
VETTORE DI TESTI — ARRAY TEXT
VETTORE DI INTERI — ARRAY INTEGER
VETTORE DI INTERI LUNGHI — ARRAY LONGINT
VETTORE DI REALI — ARRAY REAL
VETTORE DI BOOLEANI — ARRAY BOOLEAN
VETTORE DI DATE — ARRAY DATE
VETTORE DI IMMAGINI — ARRAY PICTURE
VETTORE DI PUNTATORI — ARRAY POINTER

Sintassi

VETTORE DI STRINGHE(Lunghezza;IlVettore;dim1; *dim2*)
VETTORE DI TESTI(IlVettore;dim1;*dim2*)
VETTORE DI INTERI(IlVettore;dim1;*dim2*)
VETTORE DI INTERI LUNGHI(IlVettore;dim1;*dim2*)
VETTORE DI REALI(IlVettore;dim1;*dim2*)
VETTORE DI BOOLEANI(IlVettore;dim1;*dim2*)
VETTORE DI DATE(IlVettore;dim1;*dim2*)
VETTORE DI IMMAGINI(IlVettore;dim1;*dim2*)
VETTORE DI PUNTATORI(IlVettore;dim1;*dim2*)

Nota
Le parentesi graffe sono un elemento significativo nella gestione dei vettori: sostituiremo perciò con il corsivo le graffe generalmente usate per designare un argomento opzionale.

Parametri	**Tipo**	**Descrizione**
Lunghezza	Espr_Num	Lunghezza delle stringhe unicamente per VETTORE DI STRINGHE
IlVettore	Espr_Vett	Nome della variabile vettore
dim1	Espr_Num	Numero di elementi che compongono IlVettore
dim2	Espr_Num	Numero di elementi della seconda dimensione di IlVettore

Descrizione

Tutti questi comandi creano in memoria centrale un vettore chiamato *IlVettore*. Scegliendone uno, potete specificare quale tipo di elemento volete associare a questo vettore (booleano, data, intero, intero lungo, reale, stringa, testo, immagine o puntatore). Ricordate che un vettore può contenere elementi di un solo tipo.

Durante la creazione del vettore, precisate il numero di elementi che contiene per mezzo dei parametri *Dim1* e *Dim2*. Dopo aver creato un vettore, tutti i suoi elementi vengono inizializzati da 4th DIMENSION al valore nullo.

Se il vettore esiste già quando chiamate uno di questi comandi, verranno modificate le dimensioni del vettore.
- Se impostate una dimensione inferiore alla precedente, gli elementi esclusi dalla dimensione stabilita andranno persi.
- Se impostate una dimensione superiore alla precedente, 4th DIMENSION modificherà le dimensioni del vettore, ma se non c'è memoria a sufficienza per effettuare l'operazione, il vettore resterà invariato.

Potete accedere ad un elemento del vettore indicando il nome del vettore e l'indice dell'elemento:
- Vettore ad una dimensione, assegnamento del terzo elemento:
  IlVettore{3}:="Buongiorno"
- Vettore a due dimensioni, assegnamento del quarto elemento della terza colonna
  IlVettore{3}{4}:="Buongiorno signora"

Potete ottenere il numero di elementi contenuti in un vettore usando la funzione Dimensione vettore, descritta più avanti.

I vettori sono registrati in memoria centrale. Fate attenzione a non superare la quantità di memoria disponibile: non esistono meccanismi di registrazione su disco, vale a dire che il vettore non potrà essere registrato parte in memoria centrale e parte su disco, come accade per la gestione delle schede.

Note per la Versione 1.0.x / Versione 4.x.x

- Diversamente dalla versione 1.0.x di 4th DIMENSION, non potete accedere ad un elemento del vettore con l'espressione di tipo IlVettore3, dovete necessariamente usare le parentesi graffe e scrivere IlVettore{3}.
- L'elemento 0 non contiene più la dimensione del vettore.

Esempio

VETTORE DI INTERI(Vett;10) Vett{1}:=45	Dichiarazione di un vettore di interi di 10 elementi Assegnamento dell'elemento 1
VETTORE DI INTERI(Vettore;10;10) Vett{3}{5}:=65	Dichiarazione di un vettore di interi a due dimensioni di 10 elementi per 10 Assegnamento dell'elemento 5 della terza colonna
VETTORE DI STRINGHE(5;Stringhe;10;10)	Dichiarazione di un vettore a due dimensioni di alfanumerici di 5 caratteri di 10 elementi per 10, dove Stringhe è il nome del vettore

Riferimenti

Dimensione vettore

Dimensione vettore — Size of array

Sintassi
Dimensione vettore(Vettore)
➤ Intero lungo

Parametri	Tipo	Descrizione
IlVettore	Espr_Vett	Vettore di cui si vuole conoscere la dimensione

Descrizione
Dimensione vettore restituisce il numero di elementi contenuti nel vettore *IlVettore*.

Esempio

x := **Dimensione vettore**(IlVettore)	restituisce il numero di elementi di IlVettore
y:=**Dimensione vettore**(Vettore2D{1})	restituisce il numero di elementi della colonna 1 del Vettore2D
z:=IlVettore{**Taille tableau**(IlVettore)}	restituisce il valore dell'ultimo elemento di IlVettore

Riferimenti
Comandi di dichiarazione vettori, INSERISCI ELEMENTI IN VETTORE, CANCELLA ELEMENTI VETTORE

VETTORE IN SELEZIONE — ARRAY TO SELECTION

Sintassi
VETTORE IN SELEZIONE(Vettore1;Campo1;*Vettore2*;*Campo2*;...)

Parametri	Tipo	Descrizione
Vettore1	Espr_Vett	Vettore da trasferire in un archivio
Campo1	Espr_Campo	Campo che riceve i valori del vettore

Descrizione
Il comando VETTORE IN SELEZIONE trasferisce i valori di un vettore qualsiasi alla selezione corrente dell'archivio di *Campo1*. Le schede verranno quindi aggiornate e nuovamente registrate.
- Se il numero di schede della selezione corrente è maggiore del numero di elementi di *Vettore1*, verranno aggiornate soltanto le prime n schede della selezione (n indica la dimensione del vettore).
- Se la dimensione di *Vettore1* è maggiore del numero di schede della selezione corrente, VETTORE IN SELEZIONE aggiorna le schede della selezione, poi crea quelle supplementari necessarie.
Questo comando modifica ovviamente la selezione corrente degli archivi dei campi passati come argomento. Se usate questo comando in multiutenza e se alcune schede sono in uso quando chiamate il comando VETTORE IN SELEZIONE, queste non verranno nuovamente registrate.

Esempio
Trasferimento del vettore vettNome nel campo [Clienti]Nome

VETTORE IN SELEZIONE(vettNome;[Clienti]Nome)

Trasferimento del vettore vettClienti nell'archivio Clienti
La prima colonna contiene i nomi dei clienti
La seconda colonna contiene il nome della persona da contattare nella società.

VETTORE IN SELEZIONE(vettClienti{1};[Clienti]Società;vettClienti{2};[Clienti]Contatto)

Riferimenti
SELEZIONE IN VETTORE

SELEZIONE IN VETTORE — SELECTION TO ARRAY

Sintassi
SELEZIONE IN VETTORE(Campo1;Vettore1;*Campo2*;*Vettore2*;...)

Parametri	**Tipo**	**Descrizione**
Campo1	Espr_Campo	Campo da trasferire in un vettore
Vettore1	Espr_Vett	Vettore che riceve i valori della selezione

Descrizione
SELEZIONE IN VETTORE trasferisce la selezione corrente dell'archivio di *Campo1* nel vettore *Vettore1*. La dimensione di *Vettore1* viene modificata in base al numero di elementi della selezione e aggiornata quando comparirà sullo schermo. Se uno dei vettori passati come parametro non esiste, 4th DIMENSION provvederà a crearlo.
Questo comando modifica la scheda corrente degli archivi i cui campi sono passati come parametro *Campo1...CampoN* non devono necessariamente appartenere allo stesso archivio: si può trattare di campi appartenenti ad archivi collegati per mezzo di relazioni automatiche.

Esempio

IMPOSTA ARCHIVIO CORRENTE([Clienti]) — Caricamento di un vettore a partire dall'archivio Clienti
TUTTE LE SCHEDE
SELEZIONE IN VETTORE([Clienti]Società;vettSocietà)

IMPOSTA ARCHIVIO CORRENTE[Clienti]) — Caricamento di due vettori su una relazione automatica
TUTTE LE SCHEDE
SELEZIONE IN VETTORE([Persone]Nome;vettNome;[Servizio]Nome;vettServizio)

IMPOSTA ARCHIVIO CORRENTE([Clienti]) — Caricamento di due campi in un vettore a due dimensioni
TUTTE LE SCHEDE
SELEZIONE IN VETTORE([Clienti]Nome;vettClienti{1};[Clienti]CP;vettClienti{2})

Riferimenti
VETTORE IN SELEZIONE

COPIA VETTORE — COPY ARRAY

Sintassi
COPIA VETTORE(Sorgente;Destinazione)

Parametri	Tipo	Descrizione
Sorgente	Espr_Vett	Vettore da ricopiare
Destinazione	Espr_Vett	Vettore di destinazione

Descrizione
COPIA VETTORE copia il vettore *Sorgente* nel vettore *Destinazione*. Dopo la chiamata di COPIA VETTORE, la dimensione del vettore *Destinazione* è uguale a quella del vettore *Sorgente*. Questo comando non deve essere usato per inserire elementi in un altro vettore.

Esempio

COPIA VETTORE(Vett1;Vett2)	Copia Vett1 in Vett2
COPIA VETTORE(Vett{3};Vett1)	Copia la terza colonna del vettore a due dimensioni Vett in Vett1

Riferimenti
CANCELLA ELEMENTI VETTORE, INSERISCI ELEMENTI IN VETTORE, ORDINA VETTORE

CANCELLA ELEMENTI VETTORE — DELETE ELEMENT

Sintassi
CANCELLA ELEMENTI VETTORE(IlVettore;Inizio;*NumElem*)

Parametri	Tipo	Descrizione
IlVettore	Espr_Vett	Vettore nel quale si vogliono cancellare alcuni elementi
Inizio	Espr_NumPos	Elemento a partire dal quale viene eseguita la cancellazione
NumElem	Espr_NumPos	Numero di elementi da cancellare

Descrizione
CANCELLA ELEMENTI VETTORE cancella gli elementi di *IlVettore* partendo dall'elemento *Inizio*. Se l'argomento *NumElem* viene omesso, verrà cancellata soltanto una riga. Il numero di elementi del vettore viene aggiornato automaticamente. Se *IlVettore* è un vettore a due dimensioni, CANCELLA ELEMENTI VETTORE effettuerà una cancellazione di colonne.

Esempio

CANCELLA ELEMENTI VETTORE(Vett;**Dimensione Vettore**(Vett))	Cancella l'ultimo elemento del vettore
CANCELLA ELEMENTI VETTORE(Vett;Inizio;10)	Cancella dieci elementi del vettore Vett a partire dall'elemento Inizio
CANCELLA ELEMENTI VETTORE(vettClienti{1};1;10)	Cancella i primi dieci elementi della prima colonna del vettore a due dimensioni vettClienti.
CANCELLA ELEMENTI VETTORE(vettClienti;3;1)	Cancella la terza colonna di vettClienti

Riferimenti
COPIA VETTORE, INSERISCI ELEMENTI IN VETTORE, ORDINA VETTORE

INSERISCI ELEMENTI IN VETTORE — INSERT ELEMENT

Sintassi
INSERISCI ELEMENTI IN VETTORE(IlVettore;Posizione;*NumElem*)

Parametri	**Tipo**	**Descrizione**
IlVettore	Espr_Vett	Vettore in cui vengono inseriti gli elementi
Posizione	Espr_NumPos	Posizione dell'elemento di partenza
NumElem	Espr_NumPos	Numero di elementi da inserire

Descrizione
INSERISCI ELEMENTI IN VETTORE permette di inserire un numero qualsiasi di elementi in un vettore. L'argomento *NumElem* è opzionale; se viene omesso sarà inserito un solo elemento. INSERISCI ELEMENTI IN VETTORE sposta gli elementi di indice maggiore o uguale a *Posizione* e modifica la dimensione del vettore. Se *Posizione* è uguale a **Dimensione vettore**+1, l'elemento verrà inserito nell'ultima posizione del vettore, il che corrisponde ad un inserimento in coda.
Attenzione: questo comando si limita ad inserire uno o più elementi vuoti nel vettore, i valori dovrete assegnarli voi. Se *IlVettore* è un vettore a due dimensioni, INSERISCI ELEMENTI IN VETTORE effettuerà un inserimento di colonne.

Esempio

INSERISCI ELEMENTI IN VETTORE(Vett; **Dimensione vettore**(Vett)+1)Vett{**Dimensione vettore**(Vett)}:="Nuovo elemento"	Aggiungi un elemento a fine vettore Assegna l'elemento nuovo creato
INSERISCI ELEMENTI IN VETTORE(IlVettore;1;10)	Inserisci dieci elementi all'inizio del vettore
INSERISCI ELEMENTI IN VETTORE(vettClienti{2};1;10)	Inserisci dieci elementi all'inizio della seconda colonna
INSERISCI ELEMENTI IN VETTORE(vettClienti;1;1)	Inserisci una colonna all'inizio di vettClienti

Riferimenti
COPIA VETTORE, CANCELLA ELEMENTI VETTORE, ORDINA VETTORE

Cerca in vettore — Find in array

Sintassi
Cerca in vettore(IlVettore;IlValore)
➤ Intero positivo

Parametri	Tipo	Descrizione
IlVettore	Espr_Vett	Vettore nel quale verrà svolta la ricerca
IlValore	Espr	Valore da cercare nel vettore

Descrizione
Cerca in vettore cerca in *IlVettore* il valore passato come secondo parametro e restituisce il numero del primo elemento trovato. Occorre avere una corrispondenza di tipo tra gli elementi del vettore ed il valore cercato. Cerca in vettore restituisce il numero del primo elemento corrispondente nel vettore. Se non viene trovato alcun elemento, Cerca in vettore restituirà il valore -1. Attenzione: Cerca in vettore vi darà soltanto il primo elemento trovato, non esiste il concetto di selezione nei vettori.
Non potete effettuare nessuna ricerca sui vettori di immagini o di puntatori. Se desiderate svolgere una ricerca in un vettore a due dimensioni, *IlVettore* deve essere uguale ad una delle dimensioni.

Esempio

```
x:=Cerca in vettore(vettClienti;vNome)          Cerca se esiste un elemento di vettClienti uguale a vNome
Se(x#-1)                                         Se x è diverso da -1, è stato trovato un valore
    vettClienti:=x                               Attiva l'elemento trovato nella lista sullo schermo
altrimenti
    AVVISO("Questo cliente non esiste")          Nessun elemento corrisponde alla ricerca
Fine se                                          Avvisa che la ricerca non ha dato risultati
```

Riferimenti
COPIA VETTORE, CANCELLA ELEMENTI VETTORE, INSERISCI ELEMENTI IN VETTORE, ORDINA VETTORE

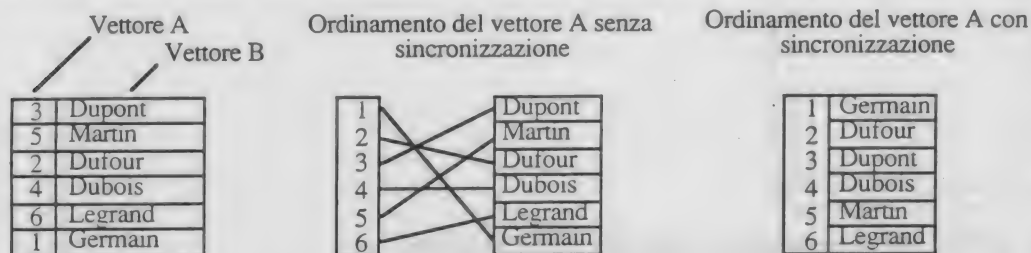
ORDINA VETTORE — SORT ARRAY

Sintassi
ORDINA VETTORE(Vettore1;*Vettore2*;...;*VettoreN*;*Ordinamento*)

Parametri	Tipo	Descrizione
Vettore1	Espr_Vett	Vettore da ordinare
Vettore2;...;VettoreN	Espr_Vett	Vettori da ordinare in base a Vett1
Ordinamento	>o<	Senso dell'ordinamento

Descrizione
ORDINA VETTORE ordina il *Vettore1* in base al senso stabilito da *Ordinamento* (crescente o decrescente).
E' possibile ordinare tutti i vettori, ad eccezione di quelli di immagini e di puntatori.
I vettori opzionali trasferiti in *Vettore2;...;VettoreN* verranno riordinati in base all'ordinamento di *Vettore1* e possono essere vettori di immagini o di puntatori.

Ordinamento del vettore A con sincronizzazione

1	Germain
2	Dufour
3	Dupont
4	Dubois
5	Martin
6	Legrand

ORDINA VETTORE(VettoreA;VettoreB;>)

Esempio

IMPOSTA ARCHIVIO CORRENTE([Clienti]) — Caricamento di due vettori e ordinamento sincronizzato
TUTTE LE SCHEDE — Seleziona tutte le schede dell'archivio Clienti
SELEZIONE IN VETTORE([Persone]Nome;vettNome;[Servizio]Nome;vettServizio)
ORDINA VETTORE(vettNome;vettServizio;>) — Ordina in modo sincronizzato i due vettori in base al vettore vettNome

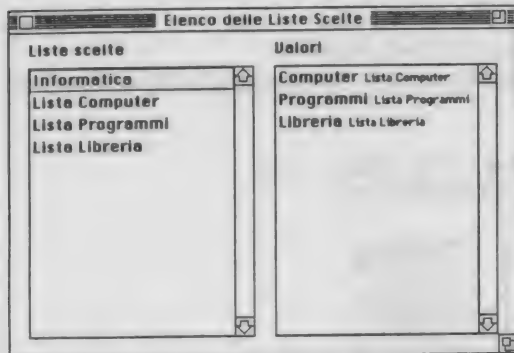
Riferimenti
COPIA VETTORE, CANCELLA ELEMENTI VETTORE, INSERISCI ELEMENTI IN VETTORE

LISTA SCELTE IN VETTORE — LIST TO ARRAY

Sintassi
LISTA SCELTE IN VETTORE(Lista;Vettore1;*Vettore2*)

Parametri	**Tipo**	**Descrizione**
Lista	Espr_Alfa	Lista scelte da cui copiare
Vettore1	Espr_Vett	Vettore in cui copiare la lista scelte
Vettore2	Espr_Vett	Vettore nel quale copiare la lista scelte collegata

Descrizione
LISTA SCELTE IN VETTORE crea il vettore *Vettore1* dalla lista scelte *Lista*, copiando i dati da *Lista* a *Vettore1*. Se *Vettore1* esiste già, il suo contenuto viene riscritto.
Il vettore Vettore2 è un argomento opzionale e viene riempito con i nomi delle liste scelte collegate.

Esempio
Supponiamo di avere le seguenti liste scelte:

LISTA SCELTE IN VETTORE("Informatica";vettLista;vettCollegate)
creerà due vettori: *vettLista*, contenente i valori "Computer->;Programmi->;Libreria->" ed il vettore *vettCollegate*, contenente i valori "Lista Computer; Lista Programmi; Lista Libreria" che sono i nomi delle scelte in relazione agli elementi della lista scelte "Informatica".

Riferimenti
VETTORE IN LISTA SCELTE

VETTORE IN LISTA SCELTE — ARRAY TO LIST

Sintassi
VETTORE IN LISTA SCELTE(Vettore1;Lista;Vettore2)

Parametri	**Tipo**	**Descrizione**
Vettore1	Espr_Vett	Vettore da copiare nella lista scelte
Lista	Espr_Alfa	Lista scelte in cui copiare il vettore
Vettore2	Espr_Vett	Vettore delle liste scelte collegate

Descrizione
VETTORE IN LISTA SCELTE copia il vettore *Vettore1* nella lista scelte *Lista*. Come per il comando LISTA SCELTE IN VETTORE, *Vettore2* è un argomento opzionale, e permette di impostare le liste scelte collegate con *Lista*.
Se *Lista* non esiste, VETTORE IN LISTA SCELTE non esegue alcuna azione.

Esempio
Vogliamo trasferire le schede dell'archiviò Famiglia in una lista scelte

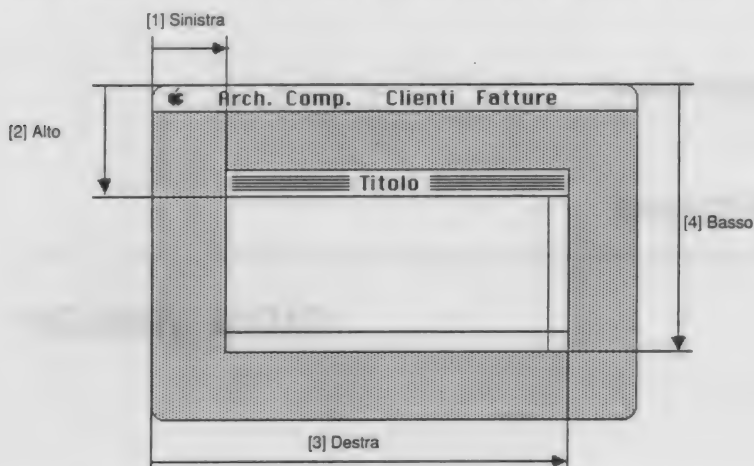
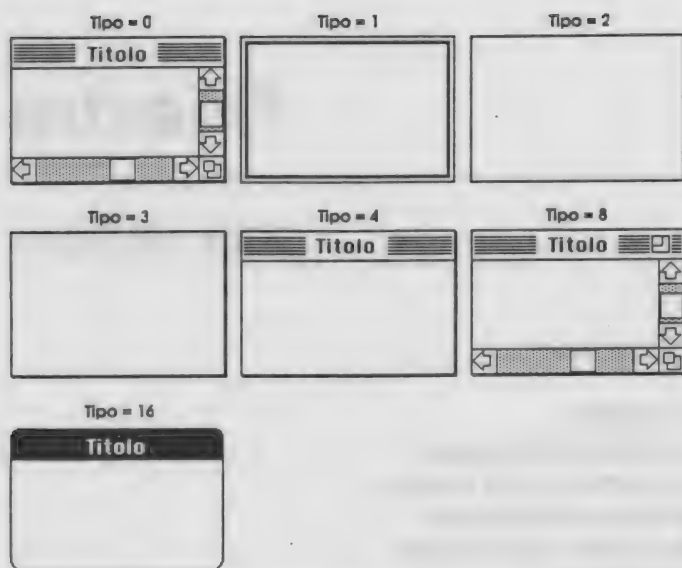
```
TUTTE LE SCHEDE([Famiglia])                          Seleziona tutte le schede dell'archivio Famiglia
SELEZIONE IN VETTORE([Famiglia]Tit;vettFamiglia)     Trasferisci la selezione nel vettore vettFamiglia
VETTORE IN LISTA SCELTE(vettFamiglia;"Famiglie")     Trasferisci il vettore alla lista scelte.
```

Riferimenti
LISTA SCELTE IN VETTORE

NUOVA FINESTRA — OPEN WINDOW

Sintassi
NUOVA FINESTRA(Sinistra;Alto;Destra;Basso;Tipo;Titolo)

Parametri	Tipo	Descrizione
Sinistra	Espr_Num	Lato sinistro della finestra
Alto	Espr_Num	Lato superiore della finestra
Destra	Espr_Num	Lato destro della finestra
Basso	Espr_Num	Lato inferiore della finestra
Tipo	Espr_Num[0..16]	Tipo di finestra
Titolo	Espr_Alfa	Titolo della finestra

Descrizione
NUOVA FINESTRA visualizza sullo schermo una finestra Macintosh che diventa la finestra attiva, cioé qualsiasi comando di visualizzazione, come MODIFICA SCHEDA o VISUALIZZA SELEZIONE avrà effetto su questa finestra.
Analogamente, l'ordine MESSAGGIO utilizza questa finestra per visualizzare i messaggi. I primi 4 parametri, *Sinistra, Alto, Destra, Basso,* permettono di stabilire le dimensioni della finestra creata a partire dall'angolo superiore sinistro (i valori sono espressi in punti).

Il quinto parametro, *Tipo*, permette di scegliere il tipo di finestra che volete aprire.
Il sesto parametro, *Titolo*, permette di scegliere il titolo della finestra, e viene ignorato per le finestre di tipo 1, 2 e 3 indicate qui di seguito:

Se aprite più di una finestra sullo schermo, soltanto l'ultima finestra aperta sarà attiva. Potete tuttavia selezionare le finestre già aperte, spostarle e modificarne le dimensioni.
In tal caso, il cursore assumerà la forma seguente:

Vi segnala che non siete sulla finestra attiva. Se scrivete qualcosa sulla tastiera o se selezionate un menu, 4th DIMENSION passerà automaticamente sulla finestra attiva.
Se la vostra finestra ha la barra del titolo, potete spostarla come più vi piace.

Per creare finestre centrate sullo schermo, consultate il comando Larghezza schermo descritto in questa stessa sezione.

Esempio

NUOVA FINESTRA(40;40;150;200;16;"FinestraCampione")
NUOVA FINESTRA(50;80;250;280;1)

Riferimenti
CHIUDI FINESTRA, PULISCI FINESTRA, MESSAGGIO

CHIUDI FINESTRA — CLOSE WINDOW

Sintassi
CHIUDI FINESTRA

Descrizione
CHIUDI FINESTRA chiude l'ultima finestra creata con il comando NUOVA FINESTRA. Se nessuna finestra utente risulta aperta, CHIUDI FINESTRA non esegue alcuna azione.

Riferimenti
NUOVA FINESTRA

CAMBIA TITOLO FINESTRA — SET WINDOW TITLE

Sintassi
CAMBIA TITOLO FINESTRA(NuovoTitolo)

Parametri	Tipo	Descrizione
NuovoTitolo	Espr_Alfa	Titolo della finestra

Descrizione
CAMBIA TITOLO FINESTRA cambia il titolo dell'ultima finestra creata con il comando NUOVA FINESTRA. Se non esiste nessuna finestra utente aperta, CAMBIA TITOLO FINESTRA non esegue alcuna azione.

Esempio
In questo esempio, cambiamo il titolo della finestra in funzione del numero di schede contenute nella selezione, come 4th DIMENSION fa automaticamente in ambiente **Utilizzo**.

CAMBIA TITOLO FINESTRA(Stringa(Schede in selezione)+" scheda/e selezionate")

Riferimenti
NUOVA FINESTRA

Altezza schermo — Screen height

Sintassi
Altezza schermo
➤ Intero

Descrizione
Altezza schermo restituisce in punti l'altezza dello schermo del Macintosh che state usando. Se avete collegato più schermi al vostro Macintosh, Altezza schermo restituirà l'altezza dello schermo principale (dove viene visualizzata la barra dei menu).

Esempio
Riferitevi all'esempio del comando Larghezza schermo descritto qui di seguito.

Riferimenti
Larghezza schermo

Larghezza schermo

Screen width

Sintassi
Larghezza schermo
➤ Intero

Descrizione
Larghezza schermo restituisce in punti la larghezza dello schermo del Macintosh che state usando. Se avete collegato più schermi al vostro Macintosh, Larghezza schermo restituirà l'altezza dello schermo principale (dove viene visualizzata la barra dei menu).
A titolo indicativo, diamo le larghezze ed altezze degli schermi Macintosh Plus, SE e II:

Macintosh	*Altezza*	*Larghezza*
Mac Plus	342	512
Mac SE	342	512
Mac II	480	640

Esempio
Ecco una procedura di 4th DIMENSION che utilizza i comandi Altezza schermo e Larghezza schermo per creare delle finestre centrate sullo schermo, indipendentemente da quello impiegato. La sintassi di chiamata di questa procedura sarà la seguente:
FinestraStandard(larghezza; altezza; tipo; titolo)

$sinistra:=(**Larghezza schermo**-$1)/2	Calcola le coordinate del lato sinistro
$destra:=$sinistra+$1	Calcola le coordinate del lato destro
Se($3=0)|($3=16)|($3=4)|($3=8)	Se la finestra ha la barra del titolo,
$Spostamento:=10	Spostala di dieci punti verso il basso
altrimenti	altrimenti
$Spostamento:=0	Nessuno spostamento
Fine se	
$alto:=((**Altezza schermo**-$2)/2)+10+$Spostamento	Calcola le coordinate del lato superiore
$basso:=$alto+$2	Calcola le coordinate dal lato inferiore
NUOVA FINESTRA($sinistra;$alto;$destra;$basso;$3;$4)	Crea la finestra

Per creare una finestra di tipo documento di 200 per 150 punti scrivete:
FinestraStandard(200;150;0;"La Finestra")

Riferimenti
Altezza schermo

VAI AXY

GOTO XY

Sintassi
VAI AXY(PosizioneX;PosizioneY)

Parametri	Tipo	Descrizione
PosizioneX	Espr_Num	Numero di colonna
Posizione Y	Espr_Num	Numero di riga

Contesto
Finestra aperta sullo schermo per mezzo del comando NUOVA FINESTRA

Descrizione
Su una finestra creata da NUOVA FINESTRA, VAI A XY posiziona il cursore sulla colonna *PosizioneX* e sulla riga *PosizioneY*. La stringa di caratteri successiva inviata dal comando MESSAGGIO verrà visualizzata a partire da questa posizione. VAI A XY non cancella il contenuto della finestra; se lo volete fare, usate il comando PULISCI FINESTRA.

Esempio
Vogliamo creare una procedura che visualizzi un messaggio durante l'aggiornamento dell'archivio Clienti. Il messaggio sarà "Aggiornamento scheda N°:" + numero della scheda. Anziché riscrivere ogni volta il messaggio completo, faremo in modo che il cursore venga collocato ogni volta a fine messaggio.

```
IMPOSTA ARCHIVIO CORRENTE([Clienti])
TUTTE LE SCHEDE
NUOVA FINESTRA(20;20;200;120;1)       Crea una finestra per contenere i messaggi
vMess:="Aggiornamento scheda N°:"     vMess contiene il messaggio da visualizzare
MESSAGGIO(vMess)                      Visualizza il messaggio una volta
Ciclo(i;1;Schede in archivio)         Esegui un ciclo da 1 fino al numero totale di schede
   VAI A XY(1;Lunghezza(vMess)+1)     Colloca il cursore dopo vMess
   MESSAGGIO(Stringa(i))              Visualizza il numero della scheda in corso di aggiornamento
   [Clienti]Fatturato:=0              Aggiorna la scheda
   REGISTRA SCHEDA                    Registra la scheda
   SCHEDA SUCCESSIVA                  Passa alla scheda successiva
Fine ciclo                            Fine ciclo
CHIUDI FINESTRA                       Chiudi la finestra al termine dell'aggiornamento.
```

Riferimenti
PULISCI FINESTRA, NUOVA FINESTRA, MESSAGGIO

PULISCI FINESTRA — ERASE WINDOW

Sintassi
PULISCI FINESTRA

Descrizione
PULISCI FINESTRA cancella il contenuto della finestra corrente e colloca il cursore al punto di origine (0,0). Se nessuna delle finestre create con il comando NUOVA FINESTRA è aperta sullo schermo, PULISCI FINESTRA non esegue alcuna azione.

Riferimenti
NUOVA FINESTRA, MESSAGGIO, VAI A XY

Menu

IMPOSTA BARRA MENU — MENU BAR

Sintassi
IMPOSTA BARRA MENU(NumBarra)

Parametri	Tipo	Descrizione
NumBarra	Espr_Num	Numero della barra di menu da visualizzare

Descrizione
IMPOSTA BARRA MENU permette di cambiare la barra di menu corrente. Quando avviate il programma la barra N° 1 è quella attiva. Se NumBarra non corrisponde ad una barra di menu esistente, IMPOSTA BARRA MENU non esegue alcuna azione.

Esempio

IMPOSTA BARRA MENU(2)
IMPOSTA BARRA MENU(vBarra)

Riferimenti
ABILITA SCELTA, DISABILITA SCELTA

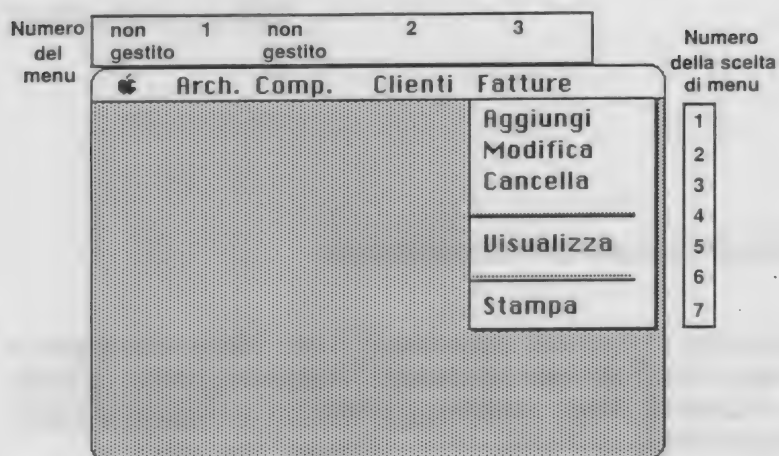
ABILITA SCELTA — ENABLE ITEM

Sintassi
ABILITA SCELTA(NumMenu;NumScelta)

Parametri	Tipo	Descrizione
NumMenu	Espr_Num	Numero del menu
NumScelta	Espr_Num	Numero della scelta di menu da attivare

Descrizione
ABILITA SCELTA rende attiva la scelta *NumScelta* del menu *NumMenu*. Se questa scelta è già attiva, ABILITA SCELTA non esegue alcuna azione. Se *NumScelta* è uguale a 0, ABILITA SCELTA attiva tutte le scelte del menu *NumMenu*.

FALSO (parz)

I menu sono numerati in questo modo:

Esempio

ABILITA SCELTA(3;2) Attiva la scelta Modifica

Riferimenti
DISABILITA SCELTA

DISABILITA SCELTA — DISABLE ITEM

Sintassi
DISABILITA SCELTA(NumMenu;NumScelta)

Parametri	Tipo	Descrizione
NumMenu	Espr_Num	Numero del menu
NumScelta	Espr_Num	Numero della scelta di menu da disabilitare

Descrizione
DISABILITA SCELTA disabilita la scelta *NumScelta* del menu *NumMenu*. Se questa scelta è già disabilitata, DISABILITA SCELTA non esegue alcuna azione. Se *NumScelta* è uguale a 0, DISABILITA SCELTA disabilita tutte le scelte del menu *NumMenu*.

Esempio

DISABILITA SCELTA(1;1) Disabilita la prima scelta del primo menu

Riferimenti
ABILITA SCELTA

Menu selezionato — Menu selected

Sintassi
Menu selezionato
➤ Intero Lungo

Contesto
Qualsiasi contesto, ad eccezione della barra di menu dello schermo principale

Descrizione
Durante l'esecuzione delle vostre procedure, se selezionate una scelta di menu, potete intercettare e identificare la scelta selezionata chiamando la funzione Menu selezionato. Questa funzione deve essere chiamata soltanto nel corso del ciclo di Durante. Menu selezionato restituisce un valore numerico composito, che potete decodificare in questo modo:

```
Se(Durante)
   Se(Menu selezionato#0)                          Se Menu selezionato è diverso da 0, significa che un
                                                    menu è stato selezionato
      MenuID:=Int(Menu selezionato/65536)           Ricava il numero del menu da questa formula
      SceltaID:=Modulo(Menu selezionato;65536)      Ricava il numero della scelta da questa formula
   Fine se
Fine se
```

Nota
Menu selezionato non vi restituisce alcun valore se avete scelto il menu Mela o il menu Composizione. Comunque, Menu selezionato permette di rilevare una scelta appartenente ad un menu associato ad un formato.

Riferimenti
IMPOSTA BARRA MENU

CONTRASSEGNA SCELTA — CHECK ITEM

Sintassi
CONTRASSEGNA SCELTA(NumMenu;NumScelta;IlSegno)

Parametri	Tipo	Descrizione
NumMenu	Espr_Num	Numero del menu
NumScelta	Espr_Num	Numero della scelta
IlSegno	Espr_Alfa	Contrassegno da visualizzare nel menu

Descrizione
CONTRASSEGNA SCELTA pone un contrassegno a lato della scelta *NumScelta* del menu *NumMenu*. Tale contrassegno è costituito solo dal primo carattere di *IlSegno*. Per cancellare un contrassegno su una scelta di menu basta chiamare CONTRASSEGNA SCELTA passando una stringa vuota in *IlSegno*.

I contrassegni sono temporanei e vengono cancellati quando si cambierà la barra di menu.
Sono utilizzati nell'interfaccia Macintosh per segnalare se un comando da menu è stato scelto o meno.
I caratteri standard di contrassegno sono quelli che potete vedere nella figura. Non è possibile ottenere questi caratteri da tastiera, si possono invece creare con il comando Carattere.

⌘ Carattere 17
✓ Carattere 18
◆ Carattere19
 Carattere 20

Nota: Non confondete i contrassegni con gli equivalenti da tastiera. I contrassegni vengono posti a sinistra del nome del comando da menu, mentre gli equivalenti da tastiera alla loro destra.

Esempio

In questo esempio creeremo un menu con 5 scelte, che permettono di cambiare la dimensione del tipo di carattere di una variabile in un formato. Vogliamo contrassegnare la dimensione attualmente selezionata.

```
Caso fra
    :(Prima)
    CONTRASSEGNA SCELTA(2;2;Carattere(18))
    vPrecedente:=2
    :(Durante)
    Se(Menu selezionato#0)
       vmenu:=Menu selezionato\65536
       vscelta:=Menu selezionato%65536
       Se(vmenu=2)
          Caso fra
             :(vscelta=1)
             CAMBIA DIMENSIONE(vtesto;9)
             :(vscelta=2)
             CAMBIA DIMENSIONE(vtesto;12)
             :(vscelta=3)
             CAMBIA DIMENSIONE(vtesto;14)
             :(vscelta=4)
             CAMBIA DIMENSIONE(vtesto;18)
             :(vscelta=5)
             CAMBIA DIMENSIONE(vtesto;20)
          Fine caso
          CONTRASSEGNA SCELTA(2;vPrecedente;"")
          CONTRASSEGNA SCELTA(2;vscelta;
          Carattere(18))
          vPrecedente:=vscelta
       Fine se
    Fine se
Fine caso
```

Contrassegna la scelta 2 del menu usando il carattere 18 (√)
Mantieni nella variabile vPrecedente, l'ultima dimensione scelta
Se qualcosa è stato modificato
Controlla se qualcosa è stato selezionato
Decodifica il menu selezionato
Se il menu selezionato è il menu nostro Stile
E' stata selezionata la scelta 1
Imposta la dimensione della variabile a 9
E' stata selezionata la scelta 2
Imposta la dimensione della variabile a 12
E' stata selezionata la scelta 3
Imposta la dimensione della variabile a 14
E' stata selezionata la scelta 4
Imposta la dimensione della variabile a 18
E' stata selezionata la scelta 5
Imposta la dimensione della variabile a 20
Togli il contrassegno della scelta precedente
Metti un contrassegno sulla nuova scelta selezionata

Registra in vPrecedente il valore della scelta attualmente selezionata

Riferimenti

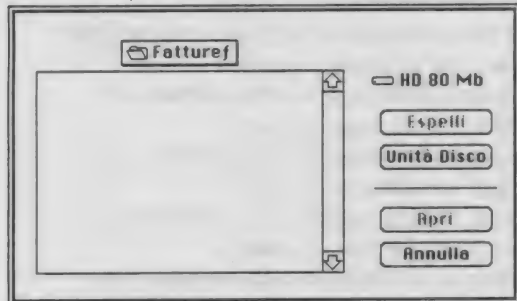
DISABILITA SCELTA, ABILITA SCELTA, IMPOSTA BARRA MENU

Apri documento

Open document

Sintassi
Apri documento(NomeDoc;{Tipo})
➤ RefNum

Parametri	**Tipo**	**Descrizione**
NomeDoc	Espr_Alfa	Nome del documento
Tipo	Espr_Alfa[4]	Tipo di documento

Descrizione
Apri documento apre il documento il cui nome è indicato da *NomeDoc*. Se *NomeDoc* è una stringa vuota, *Apri* documento presenta sullo schermo la finestra standard per l'apertura dei documenti. Se avete fatto clic sul pulsante *Apri* e l'apertura avviene correttamente, la variabile di sistema OK assumerà il valore 1, altrimenti assumerà il valore 0. Se avete confermato l'apertura, la variabile di sistema Documento conterrà il nome del documento aperto.
Potete inoltre passare il parametro *Tipo*, che vi permetterà di scegliere quale tipo di documento visualizzare nel catalogo di apertura. *Tipo* è la sigla che distingue le varie categorie di documenti su un volume. Si tratta di una stringa lunga sempre 4 caratteri. (Per esempio, la sigla standard sul Macintosh per gli archivi di tipo testo è 'TEXT').

Apri documento restituisce un valore che corrisponde al numero di riferimento del documento utilizzato dal sistema operativo del Macintosh. Questo valore verrà in seguito usato per specificare, nei comandi INVIA PACCHETTO e RICEVI PACCHETTO, a quale documento è diretta la vostra informazione.

Quando usate il comando INVIA PACCHETTO su un documento aperto per mezzo dellla funzione Apri documento, le informazioni saranno inviate a partire dall'inizio del documento. Se volete aggiungere un'informazione alla fine del documento, usate il comando Aggiungi a documento, descritto qui di seguito. Potete aprire diversi documenti contemporaneamente. In questo caso, state attenti a non superare il limite stabilito dal sistema operativo di Macintosh, che è normalmente di 12 archivi (da cui bisogna detrarne, in MultiFinder: 2 per Finder & Sistema, 1 per 4th DIMENSION, 2 per gli archivi NomeBaseDati e NomeBaseDati.data ed uno per ogni applicazione ed archivio aperto).

Il limite prestabilito di archivi aperti contemporaneamente può essere modificato; per questa operazione, rivolgetevi ad una persona competente.

Esempio

Rif1:=**Apri documento**("";"TEXT")
Rif2:=**Apri documento**("Archivio")
Rif3:=**Apri documento**("IlDisco:GliArchivi:Risultati")

Riferimenti
Aggiungi a documento, Nuovo documento, CHIUDI DOCUMENTO, INVIA PACCHETTO, RICEVI PACCHETTO

Aggiungi a documento — Append document

Sintassi
Aggiungi a documento(NomeDoc;{Tipo})
➤ RifNum

Parametri	Tipo	Descrizione
NomeDoc	Espr_Alfa	Nome del documento
Tipo	Espr_Alfa[4]	Tipo di documento

Descrizione
Aggiungi a documento effettua la stessa operazione di Apri documento, ma si posiziona a fine documento. Le informazioni saranno registrate alla fine del documento *NomeDoc*.

Esempio

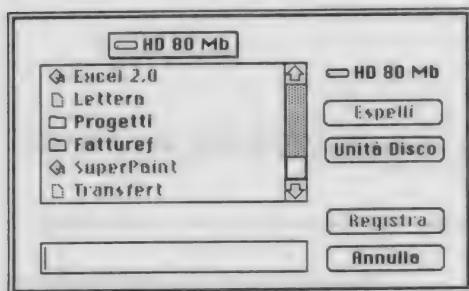
Rif1:=**Aggiungi a documento**("";"TEXT")
Rif2:=**Aggiungi a documento**("Archivio")
Rif3:=**Aggiungi a documento**("IlDisco:GliArchivi:Risultati")

Riferimenti
Apri documento, Nuovo documento, CHIUDI DOCUMENTO, INVIA PACCHETTO, RICEVI PACCHETTO

Nuovo documento — Create document

Sintassi
Nuovo documento(NomeDoc;{Tipo})
➤RifNumDoc

Parametri	Tipo	Descrizione
NomeDoc	Espr_Alfa	Nome del documento
Tipo	Espr_Alfa[4]	Tipo di documento

Descrizione
Nuovo documento crea ed apre un documento di nome NomeDoc e di tipo *Tipo*. Se *NomeDoc* è una stringa vuota, *Nuovo* documento vi proporrà la finestra standard per la creazione dei documenti. In questo caso, se fate clic sul pulsante *Registra*, la variabile di sistema OK assumerà il valore 1 e la variabile Document conterrà il nome del documento creato.

Come per Apri documento ed Aggiungi a documento, Nuovo documento restituisce il valore RifNumDoc che corrisponde al numero di archivio utilizzato dal sistema operativo del Macintosh

Esempio
Rif1:=**Nuovo documento**("Archivio";"TEXT")
Rif2:=**Nuovo documento**("";"EPSF")

Riferimenti
Apri documento, Aggiungi a documento, CHIUDI DOCUMENTO, INVIA PACCHETTO, RICEVI PACCHETTO

CHIUDI DOCUMENTO — CLOSE DOCUMENT

Sintassi
CHIUDI DOCUMENTO(NumDoc)

Parametri	**Tipo**	**Descrizione**
NumDoc	RifNumDoc	Numero del documento da chiudere

Descrizione
CHIUDI DOCUMENTO chiude il documento specificato da *NumDoc*. Se questo documento non è aperto, CHIUDI DOCUMENTO non esegue alcuna azione.
Attenzione: dovete sempre chiudere i documenti che avete precedentemente aperto o creato, altrimenti non potrete riaprirli in seguito senza riavviare il vostro Macintosh.

Riferimenti
Apri documento, Nuovo documento, Aggiungi a documento, INVIA PACCHETTO, RICEVI PACCHETTO

RICEVI PACCHETTO — RECEIVE PACKET

Sintassi
RICEVI PACCHETTO({NumDoc};StringaRicevuta;Limite)

Parametri	Tipo	Descrizione
NumDoc	RifNumDoc	Numero di riferimento del documento da cui leggere
StringaRicevuta	Espr_Var	Variabile che riceve il pacchetto
Limite	Espr_Alfa[1]	Carattere di fine ricevimento pacchetto
	o Espr_Num	Numero di caratteri da ricevere

Nota
Questo comando è inserito nella tipologia Comunicazioni della lista per tipi dei comandi nella finestra per la creazione delle procedure.

Descrizione
RICEVI PACCHETTO legge i caratteri provenienti da una porta seriale o da un documento all'interno di un volume Macintosh. Descriveremo in questo caso soltanto la lettura dei caratteri provenienti da un documento Macintosh. La descrizione di questo comando, applicato alla lettura di caratteri provenienti da una porta seriale, si trova nel capitolo relativo alle comunicazioni.

RICEVI PACCHETTO legge in un documento aperto un pacchetto di caratteri che saranno registrati nella variabile *StringaRicevuta*.
- Se l'archivio è stato aperto con il comando IMPOSTA COMUNICAZIONE, l'argomento *RifNumDoc* è superfluo, e la chiamata del comando sarà, per esempio:
RICEVI PACCHETTO(LaStringa;LaFine).

- Se l'archivio è stato aperto con i comandi Apri Documento, Aggiungi a documento o Nuovo documento, dovete passare il parametro *RifNumDoc* indicato all'apertura del documento.
Scrivete:
Rif:=**Apri Documento**("IlDocumento")
RICEVI PACCHETTO(Rif;LaStringa;LaFine)

In entrambi i casi, la lettura del pacchetto si ottiene in due modi:
- Se *Limite* è un valore numerico, RICEVI PACCHETTO leggerà un numero di caratteri uguale a *Limite*, oppure leggerà fino alla fine del documento, se il numero di caratteri da leggere è superiore alle dimensioni del documento.
- Se *Limite* è una stringa di caratteri, RICEVI PACCHETTO leggerà l'archivio finché non incontrerà una stringa uguale a *Limite*. Quando usate questa sintassi, *Limite* sarà escluso dalla stringa ricevuta.

Esempio

xRif1:=**Apri documento**("";"TEXT")	Apri un documento
Mentre(OK=1)&(**Lunghezza**([Archivio]IlTesto)<32000)	Finché non giungi alla fine del testo
RICEVI PACCHETTO(xRif1;vT;**Carattere**(13))	Leggi fino al prossimo ritorno carrello
[Archivio1]IlTesto:=[Archivio1]IlTesto+vT	Inserisci il testo letto nel campo IlTesto
Fine mentre	Chiudi il documento.
CHIUDI DOCUMENTO(xRif1)	

Riferimenti
IMPOSTA COMUNICAZIONE, Apri documento, Nuovo documento, Aggiungi a documento, INVIA PACCHETTO

INVIA PACCHETTO — SEND PACKET

Sintassi
INVIA PACCHETTO({RifDoc};StringaInviata)

Parametri	**Tipo**	**Descrizione**
RifDoc	RifNumDoc	Numero di riferimento del documento in cui scrivere
StringaInviata	Espr_Alfa	Stringa da inviare

Nota
Questo comando è inserito nella tipologia Comunicazioni della lista per tipi dei comandi nella finestra per la creazione delle procedure.

Descrizione
INVIA PACCHETTO invia i caratteri su una porta seriale o in un documento all'interno di un volume Macintosh. Descriveremo in questo caso soltanto la scrittura di caratteri verso un documento Macintosh. La descrizione di questo comando applicato all'invio di caratteri alla porta seriale si trova nel capitolo relativo alle comunicazioni.
INVIA PACCHETTO scrive in un documento aperto il pacchetto di caratteri contenuti in *StringaInviata*.
- Se l'archivio è stato aperto con il comando IMPOSTA COMUNICAZIONE, l'argomento *RifNumDoc* è superfluo, e la chiamata del comando sarà, ad esempio:
INVIA PACCHETTO(StringaInviata)
- Se l'archivio è stato aperto con i comandi Apri documento, Aggiungi a documento o Nuovo documento, dovete passare il parametro *RifNumDoc* indicato all'apertura dell'archivio.
Scrivete:
Rif:=Apri documento("IlDocumento")
INVIA PACCHETTO(Rif;StringaInviata)

Riferimenti
IMPOSTA COMUNICAZIONE, Apri documento, Nuovo documento, Aggiungi a documento, RICEVI PACCHETTO

CANCELLA DOCUMENTO | DELETE DOCUMENT

Sintassi

CANCELLA DOCUMENTO(NomeDoc)

Parametri	**Tipo**	**Descrizione**
NomeDoc	Espr_Alfa	Nome del documento Macintosh da cancellare

Descrizione

CANCELLA DOCUMENTO cancella il documento registrato con il nome *NomeDoc*. Se tale documento non esiste, CANCELLA DOCUMENTO non esegue alcuna azione. Per accedere ai documenti che non sono contenuti nella cartella della base dati, dovete specificare il percorso di accesso a questo documento, inserendolo in *NomeDoc*: "Nome del volume principale:Nome della cartella:Nome della sottocartella:...:Nome del documento".

Non potete distruggere un documento aperto. Fate comunque attenzione a non distruggere un altro documento qualsiasi, come ad esempio, una delle basi dati o un documento di sistema: potreste provocare danni irrimediabili.

Esempio

```
xRif:=Apri documento("";"TEXT")                    Apri un documento
Se(OK=1)                                            Se il documento è aperto
   Mentre(OK=1)                                     Ciclo di lettura del documento
      RICEVI PACCHETTO(V;Carattere(13))             Leggi un pacchetto
      ...                                           Sequenza di istruzioni
      ...                                           ...
   Fine mentre
   CHIUDI DOCUMENTO(xRif)                           Chiudi il documento
   CANCELLA DOCUMENTO(Documento)                    Cancella il documento
Fine se
```

Riferimenti

Apri documento, RICEVI PACCHETTO, CHIUDI DOCUMENTO

Comunicazioni

Osservazioni generali sulla comunicazione

Per stampare un documento su una stampante di tipo ImageWriter, di solito selezionate la relativa icona nella Scelta Risorse (menu Mela). Senza saperlo, in questo modo utilizzate le operazioni di comunicazione seriale del Macintosh.
In realtà, la stampante è collegata al Macintosh via cavo.
Dopo aver acceso la stampante, messo la carta, selezionato la risorsa di stampa che desiderate, avviate la stampa, partendo dalla vostra applicazione.
A questo punto, il Macintosh genera un insieme di segnali elettrici a bassa intensità in direzione della stampante. Se il cavo usato è adatto ed il driver di stampa è aggiornato, la stampante provvederà ad eseguire il lavoro richiesto.
Due sistemi, per poter comunicare fra di loro, necessitano, ovviamente, di un mezzo di collegamento.

Principi fondamentali

La trasmissione delle informazioni tra le due macchine avviene per mezzo di un cavo, attraverso il quale transitano segnali elettrici di intensità variabile.
L'insieme di queste variazioni di intensità permette di trasmettere le informazioni sulla linea codificabili in sequenze di bit (1 o 0).
In base a questo principio, le informazioni registrate inizialmente sotto forma binaria vengono trasmesse sotto forma di segnale modulato.
Per consentire alla periferica collegata al Macintosh di interpretare correttamente il segnale elettrico, il sistema ricevente deve conoscere le caratteristiche di questo segnale, ossia l'intensità e la frequenza.
Queste caratteristiche fisiche di trasmissione delle informazioni tra due sistemi vengono definite comunemente protocollo di comunicazione. In questo caso, si tratta di un protocollo di comunicazione seriale.

Le informazioni vengono trasmesse in modo sequenziale, vale a dire una per volta, e l'unità di trasmissione è il bit ("Binary Digit": entità di informazione elementare che vale 1 o 0).
Ogni carattere può essere codificato:
- su 7 bit: in questo caso, sono trasmissibili soltanto i primi 128 caratteri della tabella ASCII
- su 8 bit: in questo caso, è trasmissibile l'insieme dei caratteri. Il sistema interno di gestione delle informazioni di 4th DIMENSION avviene su 8 bit.

Supponendo di lavorare su una linea ideale, sulla quale cioè non esistano dispersioni elettriche, si potrebbe pensare che è sufficiente inviare in questo modo i caratteri da un sistema all'altro. Ora, come succede nella normale conversazione tra due individui, occorre che la persona che deve ricevere l'informazione ascolti quella che sta parlando. In effetti, se queste due persone stanno discutendo per strada, dove ci sono disturbi acustici di vario genere, l'informazione rischia di non essere compresa.

E' quindi necessario stabilire un certo numero di regole destinate a controllare il corretto svolgimento della comunicazione.
Queste regole gestiscono le operazioni di trasmissione del messaggio, e precisamente:
- l'inizio
- la trasmissione
- a scelta, l'avviso di ricezione
- la fine

Ci troviamo nel caso più semplice in cui un sistema invia informazioni ad un altro e non esistono problemi di localizzazione tra mittente e destinatario.

Le diverse fasi della comunicazione

1. L'inizio

Il sistema chiamante (o *emittente*) deve esprimere al sistema chiamato (o *destinatario*) l'intenzione d'iniziare una comunicazione.
Invia perciò un segnale, il bit di start, che avvisa il sistema chiamato di mettersi in stato di ricezione.
A partire dal momento in cui i due sistemi sono collegati fisicamente, la linea viene mantenuta in uno stato di tensione continua, cioé è percorsa da una corrente ad intensità fissa.
Questo condizionamento della linea è garantito da un insieme di componenti elettronici che si trovano all'interno di ognuno dei due sistemi.
Ritorneremo in seguito alla definizione di questi componenti.
Ognuno dei due sistemi può quindi aprire una sessione di comunicazione.

2. La trasmissione

Si tratta della trasmissione dell'informazione.
Dopo aver inviato il bit di start, il sistema chiamante può inviare una serie di 7 o 8 bit di dato, seguiti, eventualmente, da un bit di parità.
L'invio del bit di parità permette un controllo minimo sulla qualità dell'informazione ricevuta, dal momento che la tensione sulla linea potrebbe essere bassa, essere soggetta ad interferenze o in cattivo stato a causa, per esempio, di un cavo non adatto.
Potete scegliere una parità pari o dispari.
Il valore del bit di parità pari si calcola in questo modo: somma degli 1 tra i dati + il bit di parità = numero pari.

Il sistema chiamato registrerà un errore di parità quando il controllo effettuato rileverà un valore inesatto.

3. La fine

Infine, il sistema chiamante invia 1, 1.5 o 2 bit di stop, per indicare la fine del carattere.
Il numero totale di bit inviati al secondo sulla linea di trasmissione viene definito velocità di trasmissione ed è espresso in baud (baud rate).

Esistono diversi parametri che permettono di gestire la comunicazione, e precisamente:

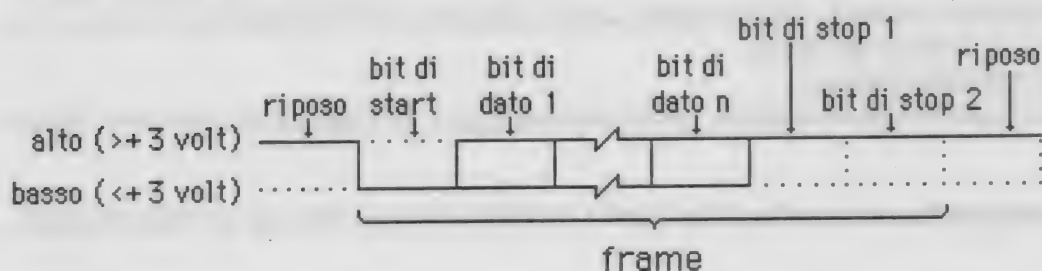
Il bit di start
Il numero dei bit di dato
Il numero dei bit di stop
La parità
La velocità

L'insieme di questi parametri vi permette di definire una comunicazione seriale.
I dati seriali vengono trasmessi da una linea di comunicazione dotata di una sola via di accesso, un bit alla volta (diversamente dai dati in parallelo, che vengono trasmessi da una linea di comunicazione a canali multipli, più bit contemporaneamente).
Il modo *Full duplex* significa che il Macintosh ed il sistema ad esso collegato possono trasmettere i dati simultaneamente (diversamente dal modo standard del Macintosh, nel quale i dati possono essere trasmessi da una sola periferica alla volta).

Il tempo che intercorre tra il bit di start e l'ultimo bit di stop viene chiamato *frame*.
Se la periferica ricevente non riceve un bit di stop dopo i bit di dato e di parità, registrerà un errore di *frame*.
Dopo i bit di stop, la periferica chiamante può inviare un altro carattere o mantenere la linea nello stato *alto*.
Se la linea viene mantenuta nello stato basso (vedi figura) per una lunghezza di frame o superiore, si verifica un *break*. I *break* vengono usati per interrompere le trasmissioni di dati.

4. La comunicazione seriale

Permette la comunicazione asincrona in modo *Full duplex*.
Esistono diverse possibilità:
- comunicazione tra due computer;
- comunicazione tra un computer ed un altro sistema.

Schema generale:

La trasmissione di dati tra due punti può assumere diversi aspetti ed è caratterizzata dalla necessità di collegare fisicamente i due punti, di stabilire un protocollo di scambio, e poi di scegliere il tipo di dati da trasmettere.

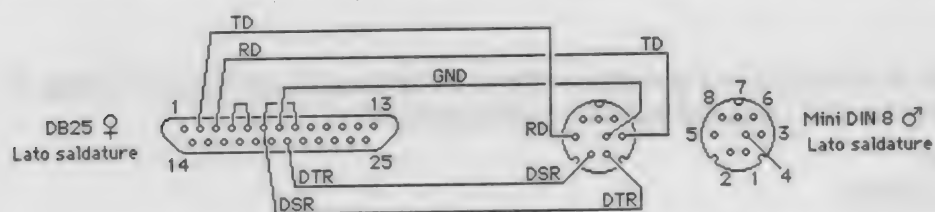
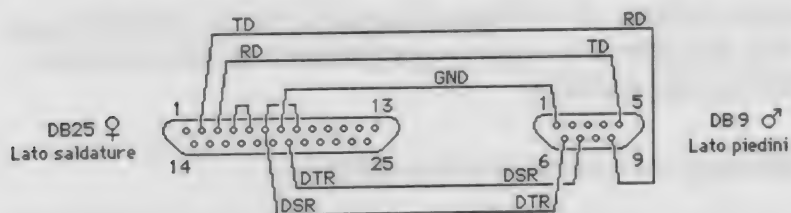
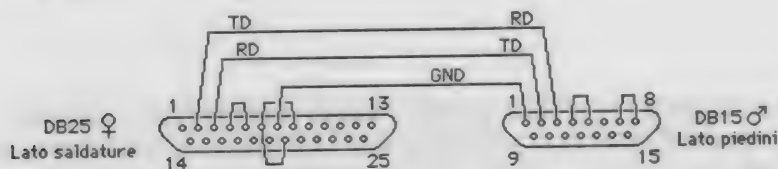
Collegamento fisico

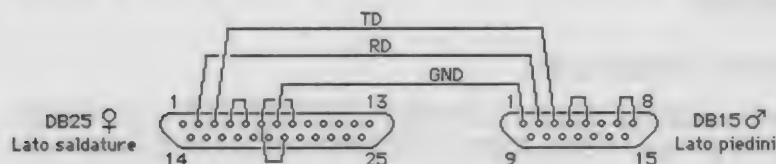
Occorre prendere in considerazione diversi casi:

1. 1. Collegamento diretto da Macintosh a Macintosh

E' sufficiente usare un cavo di collegamento Apple del tipo per stampante (Rif F590-0001) per collegare la porta seriale di un Macintosh a quella di un altro. Il cavo Modem (Rif F590-0003) è decisamente da evitare.

2. Collegamento diretto di un Macintosh ad un terminale o ad un altro computer.

Più spesso, sarà necessario realizzare un cavo come indicato nello schema seguente, che rappresenta la connessione classica tra un comune connettore RS 232 di tipo DB 25 ed il connettore Mini DIN 8 del Macintosh. Si tratta del tipo di collegamento generalmente usato fra computer e terminali.

I connettori Mini DIN 8 sono poco diffusi. Può darsi occorra anche usare un cavo adattatore Apple (rif 699-0372), che presenta un connettore Mini DIN 8 maschio ed un DB 9 femmina.
Il cavo da realizzare diventa quindi un cavo DB 25 - DB 9, con il seguente collegamento elettrico:

Alcuni terminali utilizzano i connettori DB 15. In questo caso, le connessioni da realizzare saranno le seguenti:

3. Collegamento con modem

Nel caso di collegamenti con modem, non dovrete più incrociare le connessioni TD ed RD, e neppure le connessioni DTR e DSR. Sarà sufficiente riprendere gli schemi precedenti e modificarli di conseguenza. A titolo di esempio, ecco il risultato rispetto all'ultimo schema:

Spiegazione dei codici:

GND	Ground (Massa elettrica)
TD	TransmitData (Invio dati)
RD	ReceiveData (Ricevimento dati)
DTR	Data Terminal Ready (Terminale pronto a ricevere)
DSR	Data Set Ready (Dati pronti da trasmettere)

4. Impostazioni

Non bisogna dimenticare che la trasmissione è possibile soltanto tra due apparecchiature impostate in modo coerente. Dovete assicurarvi che i seguenti parametri siano comuni:

Velocità di trasmissione:
da 300 a 19600 baud (bit/secondo)

Numero di bit
7 o 8 bit compongono il carattere trasmesso. La scelta di 7 bit non permetterà la trasmissione dei codici ASCII superiori a 127, delle schede 4th DIMENSION, delle immagini o dei suoni digitalizzati

Parità
Può essere di tipo pari o dispari. Si tratta del controllo di correttezza eseguito dalle due macchine, una delle quali completa ogni stringa di bit che rappresenta un carattere per ottenere un numero di bit a 1 sempre pari (o dispari), mentre l'altra verifica questo controllo.

Bit di stop
1, 1.5, 2, vi permettono di impostare il segnale di fine trasmissione di un carattere.

Protocolli

Esistono 3 possibilità:

Nessun protocollo
Significa che la trasmissione dei dati avviene senza controllo, ed alla velocità dell'emittente. Ad esempio, se trasmettete dati da un terminale a 4th DIMENSION, è sufficiente che il terminale liberi la memoria sulla porta seriale, il che può avvenire molto rapidamente, mentre 4th DIMENSION crea una scheda per ogni dato, riceve il dato, registra la scheda ed aggiorna i propri indici, e questa operazione è più lenta.

Dopo un certo numero di dati, il "buffer" è saturo ed i dati successivi andranno persi.
Di conseguenza, occorre stabilire uno scambio di informazioni tra l'emittente ed il ricevente, affinchè il secondo possa avvertire il primo quando è saturo, ed il primo possa attendere prima di continuare l'emissione. Questo scambio si chiama Handshaking (stretta di mano) e può assumere diverse forme.

Protocollo DTR: Data Terminal Ready
Si tratta di un Handshaking di tipo hardware: significa che le informazioni necessarie al dialogo sono trasportate da due conduttori, separati dai due conduttori di trasmissione dati. Uno dei due invia il DSR, ossia avverte che dispone di dati pronti per la trasmissione. L'altro risponde allora con un DTR, dicendo: "Sono pronto a ricevere".

Questo scambio è permanente e permette di confermare o annullare la trasmissione dei dati che circolano per mezzo dei conduttori TD ed RD. Un cavo DTR contiene un minimo di 5 conduttori; per questo motivo il protocollo non è applicabile tra due modem (collegati da una semplice linea telefonica, doppino + terra).

Protocollo XON-XOFF
Si tratta di un Handshaking di tipo software, i cui codici di conferma vengono trasmessi dai conduttori TD ed RD tra i pacchetti di dati e sono:

Codice ASCII 02: STX Start transmit (Inizio pacchetto)
Codice ASCII 03: ETX End Transmit (Fine pacchetto)
Codice ASCII 04: EOT End of transmission (Fine trasmissione)
Codice ASCII 05: ENQ Enquiry (Richiesta trasmissione)
Codice ASCII 06: ACK Acknowledge (Consenso trasmissione)

E' molto importante, quando usate il protocollo XON-XOFF, evitare di trasmettere questi codici, per non disturbare l'handshaking. E' il solo modo utilizzabile in una trasmissione con modem, dato che sono disponibili soltanto due conduttori.

Interruzione della comunicazione sulla porta seriale
Per rispettare i sistemi già esistenti, l'interruzione di una comunicazione seriale si ottiene premendo Opzione-Comando-Maiuscole

---

Tipi di dati

4th DIMENSION autorizza la trasmissione di dati secondo diverse modalità, che vengono scelte in base al protocollo.

Quando 4th DIMENSION invia delle schede, queste contengono valori di tipo reale, intero, o immagine, codificati usando tutti i valori possibili per byte, ossia da 00 a FF.

Nota:
Se volete usare l'ImageWriter in ASCII per stampare con il comando INVIA PACCHETTO, non dimenticate di usare IMPOSTA COMUNICAZIONE per impostare il protocollo in DTR e di verificare l'impostazione del microswitch 2-3 (posizione ON). Questa azione non può essere effettuata a distanza.

IMPOSTA COMUNICAZIONE — SET CHANNEL

Sintassi
IMPOSTA COMUNICAZIONE(Porta;Param)
IMPOSTA COMUNICAZIONE(Modo;NomeDoc)

Parametri	**Tipo**	**Descrizione**
Porta	Espr_Num	Scelta della porta e del modo di comunicazione
Param	Espr_Num	Parametri di comunicazione
o		
Modo	Espr_Num	Modo, lettura o scrittura
NomeDoc	Espr_Alfa	Nome del documento

Descrizione
Il comando IMPOSTA COMUNICAZIONE permette di gestire due tipi di comunicazioni:
- comunicazioni sulle porte seriali del Macintosh
- comunicazioni con un documento che si trova su un volume in linea.

Prima sintassi - Comunicazione con la porta seriale:
Il parametro Porta permette di scegliere la porta seriale ed il protocollo che verranno utilizzati.
I valori possibili sono i seguenti:

Porta	*Porta seriale*	*Protocollo usato*
0	Porta stampante	standard
1	Porta modem	standard
20	Porta stampante	XON/XOFF
21	Porta modem	XON/XOFF
30	Porta stampante	DTR(Data Terminal Ready)
31	Porta modem	DTR (Data Terminal Ready)

Param permette di scegliere i parametri della comunicazione: velocità, numero dei bit di dato, tipo di parità, bit di stop.

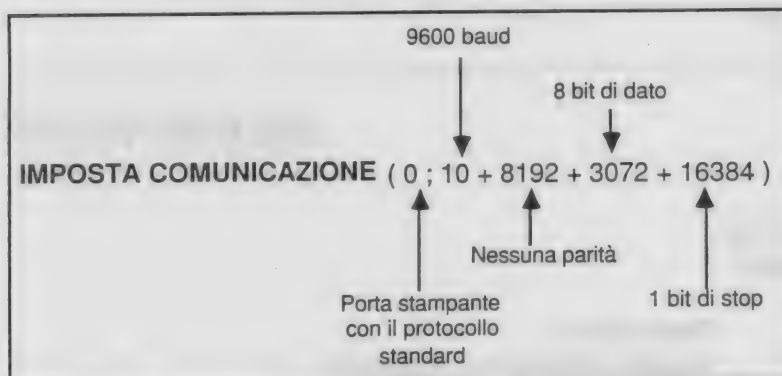
Velocità	*Valore*
300 baud	380
600 baud	189
1200 baud	94
1800 baud	62
2400 baud	46
3600 baud	30
4800 baud	22
7200 baud	14

9600 baud	10
19200 baud	4
57000 baud	0

Num di bit di dato	*Valore*
5 bit	0
6 bit	2048
7 bit	1024
8 bit	3072

Parità	*Valore*
Nessuna parità	8192
Parità dispari	4096
Parità pari	12288

Nr di bit di stop	*Valore*
1	16384
1,5	-32768
2	-16384

Esempio di impostazione
Per impostare la porta stampante con il protocollo standard, alla velocità di 9600 baud, senza parità, a 8 bit di dato con 1 bit di stop, occorre scrivere

Seconda sintassi - Operazione su un documento Macintosh:
IMPOSTA COMUNICAZIONE permette di creare, aprire e richiudere un documento Macintosh registrato su disco. *Modo* indica il tipo di operazione da effettuare, *NomeDoc* deve essere un'espressione contenente il nome di un documento.

Apertura di un documento qualsiasi:
IMPOSTA COMUNICAZIONE(10;*NomeDoc*)
Con questa sintassi, IMPOSTA COMUNICAZIONE apre il documento Macintosh specificato da *NomeDoc*. Se il documento esiste, viene aperto, altrimenti 4th DIMENSION lo crea e poi lo apre.
Con questa sintassi, le variabili di sistema OK e Document non restituiscono valori significativi.

Apertura di un documento di tipo 'TEXT':
IMPOSTA COMUNICAZIONE(13;"")
Se passate 13 come parametro in *Modo* dovete necessariamente passare una stringa vuota in *NomeDoc*.
Con questa sintassi, IMPOSTA COMUNICAZIONE visualizza sullo schermo la finestra standard per l'apertura dei documenti. Se aprite un documento, la variabile OK restituisce il valore 1 e la variabile Document restituisce il nome del documento aperto, altrimenti OK restituisce 0.

Creazione di un documento su disco:
IMPOSTA COMUNICAZIONE(12;"")
Se passate 12 come parametro in *Modo*, dovete necessariamente passare una stringa vuota in *NomeDoc*. Con questa sintassi, IMPOSTA COMUNICAZIONE visualizza sullo schermo la finestra standard per l'apertura dei documenti. Se create (e, di conseguenza, aprite) un documento, la variabile OK restituisce il valore 1 e la variabile Document restituisce il nome del documento aperto, altrimenti OK restituisce 0.

Chiusura di un documento:
IMPOSTA COMUNICAZIONE(11)
Se passate 11 come parametro in *Modo*, non dovrete passare un secondo parametro. Con questa sintassi, IMPOSTA COMUNICAZIONE chiude un documento aperto in precedenza.
Attenzione: dovete sempre chiudere i documenti precedentemete aperti o creati, altrimenti questi non potranno essere riaperti senza riavviare il Macintosh.

Riferimenti
INVIA PACCHETTO, RICEVI PACCHETTO

RICEVI PACCHETTO — RECEIVE PACKET

Sintassi
RICEVI PACCHETTO(LaVariabile;NumCar)
RICEVI PACCHETTO(LaVariabile;Limite)

Parametri	Tipo	Descrizione
LaVariabile	Espr_Var	Variabile che riceve il pacchetto
NumCar	Espr_Num	Numero di caratteri da ricevere
Limite	Espr_Alfa[1]	Carattere di fine ricevimento pacchetto

Descrizione
RICEVI PACCHETTO legge i dati provenienti dalle porte seriali o da un documento aperto con il comando IMPOSTA COMUNICAZIONE. Fate riferimento al capitolo Documenti Macintosh per la sintassi del comando RICEVI PACCHETTO applicata ai documenti Macintosh.

RICEVI PACCHETTO ammette 2 sintassi, diverse fra loro nel modo di delimitare l'informazione ricevuta.
- La prima sintassi leggerà un numero di caratteri pari a *NumCar*, introducendo il risultato nella variabile *LaVariabile*
- La seconda sintassi di RICEVI PACCHETTO leggerà l'informazione finché non viene ricevuto il primo carattere di *Limite*. I caratteri ricevuti vengono introdotti nella variabile *LaVariabile*, ad eccezione del carattere *Limite*.

Se ricevete dei caratteri attraverso la porta seriale, essi possono provenire da un'altra applicazione di 4th DIMENSION o da un altro programma, o persino da un altro computer in grado di inviare informazioni attraverso l'interfaccia RS-232.

Esistono due modi per interrompere una comunicazione seriale:
- premendo Opzione-Barra spaziatrice
- superando il timeout stabilito dal comando IMPOSTA TIMEOUT.

Potete controllare se la comunicazione è terminata correttamente verificando la variabile di sistema OK, che assumerà il valore 1. Se avete interrotto la comunicazione o se si sono verificati degli errori di comunicazione, la variabile OK assumerà il valore 0.

Esempio

```
IMPOSTA ARCHIVIO CORRENTE([Archivio1])              Ricevi sulla porta seriale e registra le informazioni
IMPOSTA COMUNICAZIONE(1;10+8192+3072+16384)         Imposta i parametri di comunicazione
Ripeti                                              Ripeti la sequenza fino al ricevimento di "Fine"
   RICEVI PACCHETTO(vIn;Carattere(13))              Attendi un pacchetto la cui fine è un ritorno carrello
   Se(vIn#"Fine")                                   Se il pacchetto ricevuto non è "Fine"
      NUOVA SCHEDA                                  Crea una scheda
      [Archivio1]Campo1:=vIn                        Registra il valore ricevuto nel campo
      REGISTRA SCHEDA                               Registra la scheda
   Fine se
Finché((OK=0)|(vIn="Fine"))
```

Riferimenti
INVIA PACCHETTO, IMPOSTA COMUNICAZIONE

INVIA PACCHETTO — SEND PACKET

Sintassi
INVIA PACCHETTO(LaStringa)

Parametri	Tipo	Descrizione
LaStringa	Espr_Alfa	Informazione da inviare

Descrizione
INVIA PACCHETTO invia *LaStringa* sulla porta seriale o nel documento aperto con il comando IMPOSTA COMUNICAZIONE.

Potete utilizzare questo comando per inviare informazioni ad un altro 4th DIMENSION, ad un altro programma per Macintosh, ad un altro computer, ad una stampante dotata di interfaccia seriale (tipo ImageWriter), ad un videodisco o a qualsiasi apparecchiatura dotata di una porta seriale di tipo RS-232.

Se inviate informazioni tramite la porta seriale, noterete che i caratteri Macintosh sono codificati su 8 bit (per gli accenti ed i caratteri speciali); alcuni computer e periferiche supportano soltanto 7 bit di dato; in questo dovrete servirvi di un filtro di conversione ASCII per poter convertire i caratteri con codici ASCII superiori a 127.

Potete interrompere una comunicazione seriale in due modi:
- premendo Opzione-Barra spaziatrice
- superando il timeout stabilito dal comando IMPOSTA TIMEOUT

Esempio

```
IMPOSTA ARCHIVIO CORRENTE([Archivio1])
TUTTE LE SCHEDE                                      Seleziona tutte le schede
IMPOSTA COMUNICAZIONE(1;10+8192+3072+16384)          Imposta i parametri di comunicazione
Mentre(Non(Fine selezione))                          Finché non raggiungi la fine dell'archivio
   INVIA PACCHETTO([Archivio1]Campo1+Carattere(13))  Invia il contenuto del campo 1 seguito da un
   SCHEDA SUCCESSIVA                                 carattere di separazione
Fine mentre                                          Passa alla scheda successiva
```

Riferimenti
RICEVI PACCHETTO, IMPOSTA COMUNICAZIONE

INVIA VARIABILE — SEND VARIABLE

Sintassi
INVIA VARIABILE(LaVariabile)

Parametri	Tipo	Descrizione
LaVariabile	Espr_Var	Variabile da inviare

Descrizione
INVIA VARIABILE invia *La variabile* attraverso la porta seriale aperta con il comando IMPOSTA COMUNICAZIONE o nel documento selezionato con il comando IMPOSTA COMUNICAZIONE.
In entrambi i casi, il programma che riceve l'informazione deve essere un altro 4th DIMENSION, in quanto le variabili vengono inviate nel loro formato interno che ne comprende il valore e il tipo, formato che può essere letto soltanto se usate il comando RICEVI VARIABILE.
Una comunicazione seriale può essere interrotta in due modi:
- premendo Opzione-Barra spaziatrice
- superando il timeout stabilito dal comando IMPOSTA TIMEOUT.

Nota per la Versione 1.0.x / Versione 4.x.x
Il formato per l'invio delle variabili usato dal comando RICEVI VARIABILE nella versione 1.0.x è diverso da quello usato nella versione 4.x.x, di conseguenza le informazioni non sono compatibili tra di loro.

Esempio
Fate riferimento all'esempio del comando INVIA SCHEDA descritto qui di seguito.

Riferimenti
RICEVI VARIABILE, INVIA SCHEDA

INVIA SCHEDA — SEND RECORD

Sintassi
INVIA SCHEDA({Archivio})

Parametri	Tipo	Descrizione
Archivio	Espr_Arch	Archivio della scheda da inviare

Descrizione
INVIA SCHEDA invia la scheda corrente di *Archivio* alla porta seriale selezionata con il comando IMPOSTA COMUNICAZIONE o al documento apertO con il comando IMPOSTA COMUNICAZIONE. L'argomento *Archivio* è opzionale. Se questo argomento viene omesso, INVIA SCHEDA si riferirà all'archivio corrente. Se *Archivio* non ha un scheda corrente, INVIA SCHEDA non esegue alcuna azione.
Le schede vengono inviate da INVIA SCHEDA in un formato interno di 4th DIMENSION, di conseguenza potranno essere rilette soltanto dal comando RICEVI SCHEDA.
Se volete inviare schede ad un'altra applicazione, usate i comandi INVIA PACCHETTO o ESPORTA TESTO.
Una comunicazione seriale può essere interrotta in due modi:
- premendo Opzione-Barra spaziatrice
- superando il timeout stabilito dal comando IMPOSTA TIMEOUT.

Nota per la Versione 1.0.x / Versione 4.x.x
Il formato per l'invio delle schede usato dal comando RICEVI SCHEDA nella versione 1.0.x è diverso da quello della versione 4.x.x. Di conseguenza, le informazioni non sono compatibili fra di loro.

Esempio
Vogliamo creare un documento di archiviazione; per questo motivo creiamo un documento ed usiamo i comandi INVIA SCHEDA e INVIA VARIABILE.

```
IMPOSTA ARCHIVIO CORRENTE([Clienti])        Crea il documento
IMPOSTA COMUNICAZIONE(12;"Archivio")        Se la creazione del documento è confermata
Se(OK=1)                                    Seleziona tutte le schede
   TUTTE LE SCHEDE
   Ripeti                                   Crea una variabile contenente il numero della scheda
      vRif:=[Clienti]Riferimento            Invia la variabile vRif
      INVIA VARIABILE(vRif)                 Invia la scheda
      INVIA SCHEDA                          Passa alla scheda successiva
      SCHEDA SUCCESSIVA                     Finché non giungi al termine della selezione
   Finché(Fine selezione)                   Invia una variabile contenente "Fine" per segnalare la fine
   vRif:="Fine"                             dell'archivio
   INVIA VARIABILE(vRif)
Fine se                                     Chiudi l'archivio
IMPOSTA COMUNICAZIONE(11)
```

Riferimenti
INVIA VARIABILE, RICEVI SCHEDA

RICEVI VARIABILE — RECEIVE VARIABLE

Sintassi
RICEVI VARIABILE(LaVariabile)

Parametri	**Tipo**	**Descrizione**
LaVariabile	Espr_Var	Variabile da ricevere

Descrizione
RICEVI VARIABILE attende e riceve *LaVariabile* proveniente dalla porta seriale precedentemente selezionata con il comando IMPOSTA COMUNICAZIONE o da un documento aperto con IMPOSTA COMUNICAZIONE. RICEVI VARIABILE riceve una variabile nel formato interno di 4th DIMENSION: la lettura perciò può provenire solo da un altro programma 4th DIMENSION che invia una variabile con il comando INVIA VARIABILE o da un documento creato da un'applicazione di 4th DIMENSION.

Una comunicazione seriale può essere interrotta in due modi:
- premendo Opzione-Barra spaziatrice
- superando il timeout stabilito dal comando IMPOSTA TIMEOUT
Potete controllare questa interruzione verificando il valore della variabile di sistema OK, che in questo caso sarà uguale a 0. Se la comunicazione si è svolta correttamente, OK assumerà il valore 1.

Nota per la Versione 1.0.x / Versione 4.x.x
Il formato utilizzato dal comando RICEVI VARIABILE nella versione 1.0.x è diverso da quello della versione 4.x.x, di conseguenza le informazioni non sono compatibili tra di loro.

Esempio
Fate riferimento all'esempio del comando RICEVI SCHEDA

Riferimenti
INVIA VARIABILE, RICEVI SCHEDA

RICEVI SCHEDA — RECEIVE RECORD

Sintassi
RICEVI SCHEDA({Archivio})

Parametri	Tipo	Descrizione
Archivio	Espr_Arch	Archivio che conterrà la scheda da ricevere

Descrizione
RICEVI SCHEDA riceve tutte le informazioni di una scheda inviate attraverso la porta seriale o provenienti da un documento Macintosh aperto con il comando IMPOSTA COMUNICAZIONE, ed introduce il risultato nella scheda corrente di *Archivio*. *Archivio* è opzionale. Se questo argomento viene omesso, RICEVI SCHEDA si riferirà all'archivio corrente. Se ricevete questa scheda in una già esistente, i valori precedenti dei campi verranno sostituiti dai valori ricevuti. Se ricevete queste informazioni in una nuova scheda, dovrete precedentemente chiamare il comando NUOVA SCHEDA prima di RICEVI SCHEDA.
In entrambi i casi, dovete chiamare esplicitamente il comando REGISTRA SCHEDA se volete conservare la scheda ricevuta.
Le schede ricevute non devono necessariamente presentare la stessa struttura di Archivio: 4th DIMENSION cercherà di convertire i campi della scheda ricevuta in base alla struttura dell'archivio a cui è destinata.

Sia che riceviate la scheda da un documento o da una porta seriale, essa dovrà rispettare il formato interno di 4th DIMENSION e perciò potrà provenire unicamente da un'applicazione di 4th DIMENSION che invia la scheda usando il comando INVIA SCHEDA.
Una comunicazione seriale può essere interrotta in due modi:
- premendo Opzione-Barra spaziatrice
- superando il timeout stabilito dal comando IMPOSTA TIMEOUT. Potete controllare questa interruzione verificando il valore della variabile di sistema OK, che sarà uguale a 0. Se la comunicazione si è svolta correttamente, OK assumerà il valore 1.

Nota per la Versione 1.0.x / Versione 4.x.x
- Il formato per l'invio delle schede utilizzato dal comando RICEVI SCHEDA nella versione 1.0.x è diverso da quello della versione 4.x.x, di conseguenza le informazioni non sono compatibili tra di loro.
- Diversamente dalla versione 1.0.x di 4th DIMENSION, non siete obbligati ad usare una comunicazione su 8 bit di dato, potete servirvi di tutti i parametri disponibili (da 5 a 8 bit di dato), ed anche del protocollo XON/XOFF.

Esempio
Questa procedura vi permetterà di rileggere il documento di archiviazione creato nell'esempio del comando INVIA SCHEDA.

```
IMPOSTA ARCHIVIO CORRENTE([Archivio])
IMPOSTA COMUNICAZIONE(10;"")              Se l'apertura del documento è confermata
Se(OK=1)
   Ripeti
      RICEVI VARIABILE(vRif)              Leggi la variabile all'inizio della scheda
      Se(OK=1)&(vRif#"Fine")              Se non è uguale a "Fine"
         NUOVA SCHEDA([Archivio])         Crea una scheda nella memoria
         RICEVI SCHEDA                    Leggi la scheda
         REGISTRA SCHEDA                  Registra la scheda
      Fine se
   Finché(OK=0)|(vRif="Fine")             Ripeti finchè Rif non diventa uguale a "Fine"
Fine se
IMPOSTA COMUNICAZIONE(11)                 Chiudi l'archivio
```

Riferimenti
INVIA SCHEDA, RICEVI VARIABILE, RICEVI PACCHETTO

RICEVI BUFFER — RECEIVE BUFFER

Sintassi
RICEVI BUFFER(IlBuffer)

Parametri	Tipo	Descrizione
IlBuffer	Espr_Var	Variabile che riceve il contenuto del buffer associato alla porta seriale

Contesto
Esclusivamente nel caso di comunicazione seriale

Descrizione
RICEVI BUFFER prende il contenuto del buffer dalla porta seriale aperta con IMPOSTA COMUNICAZIONE, inserendolo nella variabile *IlBuffer* e libera il buffer associato alla porta seriale. Diversamente dal comando RICEVI PACCHETTO, RICEVI BUFFER non blocca l'esecuzione del programma fino al ricevimento di un certo numero di caratteri. Potete usarlo con il comando CHIAMA SU PORTA SERIALE.

Esempio
Supponiamo di aver creato una base dati nella quale potete ricevere in qualsiasi momento le informazioni provenienti da altri utenti, attraverso la porta seriale.
Innanzi tutto, userete una procedura per avvertire 4th DIMENSION di prendere in considerazione tutte le informazioni ricevute sulla porta seriale che volete utilizzare.

```
IMPOSTA COMUNICAZIONE(1;10+8192+3072+     Scegli i parametri per la porta seriale utilizzata, Porta
16384)                                    Modem, 9600 baud
RICEVI BUFFER(t)                          Libera il buffer della seriale da eventuali altri caratteri
CHIAMA SU PORTA SERIALE("Chi")            Installa la procedura di interruzione.
```

Al termine di questa procedura, 4th DIMENSION eseguirà la procedura *Chi* non appena il Macintosh riceverà informazioni sulla porta seriale. La procedura *Chi* vi permetterà di conoscere il nome dell'interlocutore che vi sta chiamando (avete precedentemente definito un protocollo nel quale ogni comunicazione inizia con l'identificazione della persona che vi chiama). Ora, siccome il nome di questa persona presenta un numero di caratteri variabile, e non ritenete opportuno usare un delimitatore, dovrete programmare la procedura *Chi* in questo modo:

```
$on:=1
$Res:=""
Mentre($on=1)
    RICEVI BUFFER(v)
    Se(v="")
        $on:=0
    Fine se
    $Res:=$Res+v
Fine mentre
Vnome:=$Res
AVVISO($Res+"avvia una sessione di comunicazione")
```

Riferimenti
RICEVI PACCHETTO, CHIAMA SU PORTA SERIALE

USA FILTRO ASCII — USE ASCII MAP

Sintassi
USA FILTRO ASCII(IlFiltro|*;{TipoFiltro})

Parametri	Tipo	Descrizione
IlFiltro	Espr_Alfa	Nome del filtro ASCII da utilizzare
*	Costante	Opzione di caricamento del filtro standard del Macintosh
TipoFiltro	Espr_Num	Filtro per l'importazione o per l'esportazione

Descrizione
USA FILTRO ASCII carica in memoria il filtro ASCII precedentemente registrato con il nome di *IlFiltro.* Tale filtro potrà essere creato in ambiente Utilizzo, con i comandi Filtro ASCII per Import..., Filtro ASCII per Export...
Se *TipoFiltro* è uguale a 0 o se questo argomento viene omesso, il filtro sarà utilizzato per le operazioni di esportazione.
Se *TipoFiltro* è uguale a 2, IlFiltro sarà utilizzato per le operazioni di importazione.
Una volta caricato in memoria, verrà utilizzato da 4th DIMENSION per tutte le operazioni di trasferimento tra 4th DIMENSION e l'esterno, per i dati trasferiti con i comandi d'importazione-esportazione ASCII, SYLK e DIF, per quelli inviati con i comandi INVIA PACCHETTO, RICEVI PACCHETTO, RICEVI BUFFER. I filtri non hanno nessun effetto sui dati trasferiti per mezzo dei comandi INVIA SCHEDA, INVIA VARIABILE, RICEVI SCHEDA, RICEVI VARIABILE.
Al termine delle operazioni, potrete ripristinare il filtro ASCII originario, scrivendo USA FILTRO ASCII(*).

Esempio

USA FILTRO ASCII("FiltroExcel")	Installa il filtro "FiltroExcel"
ESPORTATESTO([Clienti];"RapportoFatturato89")	Esporta l'archivio [Clienti]
USA FILTRO ASCII(*)	Ritorna al filtro standard

Riferimenti
IMPORTA ESPORTA SYLK, DIF, ASCII, INVIA PACCHETTO, RICEVI PACCHETTO

IMPOSTA TIMEOUT — SET TIMEOUT

Sintassi
IMPOSTA TIMEOUT(Tempo)

Parametri	Tipo	Descrizione
Tempo	Espr_Num	Durata in secondi

Contesto
Comunicazione seriale

Descrizione
IMPOSTA TIMEOUT permette di impostare il tempo di attesa massimo quando usate i comandi RICEVI PACCHETTO, RICEVI VARIABILE, RICEVI SCHEDA, INVIA PACCHETTO, INVIA VARIABILE, INVIA SCHEDA nelle comunicazioni seriali. Se dopo un certo numero di secondi uguale a Tempo non avete ricevuto nulla, 4th DIMENSION interromperà la comunicazione, come avreste potuto fare premendo Command-Barra spaziatrice.

Esempio

```
IMPOSTA COMUNICAZIONE(1;10+8192+3072+16384)    Imposta i parametri di comunicazione
IMPOSTA TIMEOUT(10)                            Interrompi il collegamento se non succede nulla
                                               dopo 10 secondi
Ripeti                                         Ripeti la seguente sequenza
   RICEVI PACCHETTO(v;Carattere(13))           Aspetta di ricevere un pacchetto di caratteri
   Se(OK=0)                                    Se OK è uguale a 0, c'è un problema di comunica-
      CONFERMA("Problema di comunicazione, vuoi zione. Chiedi la conferma per continuare
      continuare?")
   Fine se
Finché(OK=0)|(v="Fine")
```

Riferimenti
IMPOSTA COMUNICAZIONE, RICEVI PACCHETTO, RICEVI SCHEDA, RICEVI VARIABILE

Descrizione struttura

Numero di archivi — Count files

Sintassi
Numero di archivi
➤ Intero positivo

Descrizione
Numero di archivi restituisce il numero di archivi della base dati attualmente aperta.

Esempio
AVVISO("La base dati contiene "+**Stringa**(**Numero di archivi**;"##0")+" archivio/i")

Riferimenti
Numero di campi, ATTRIBUTI CAMPO, Nome del campo, Nome archivio

Numero di campi — Count fields

Sintassi
Numero di Campi(Archivio_Num)
➤Intero positivo

o

Numero di campi(Puntatore_Archivio)
➤ Intero positivo

Parametri	Tipo	Descrizione
Archivio_Num	Espr_NumPos	Numero interno dell'archivio
Puntatore_Archivio	Punt_Archivio	Puntatore di un archivio

Descrizione
Numero di campi restituisce un intero positivo pari al numero di campi definiti nell'archivio passato come parametro. Potete far riferimento all'archivio usando il suo numero (ordine di creazione degli archivi) o con un puntatore all'archivio.
Quando passate un puntatore ad un archivio, dovete passare come parametro il puntatore (»[Archivio1]) e non il valore puntato.
Se R=»[Archivio1] non bisogna passare R», ma soltanto R.

Esempio

Num_Campi:=**Numero di Campi**(1)
Num_Campi:=**Numero di Campi**(»[Archivio1])
Rif:=»[Archivio1]
Num_Campi:=**Numero di Campi**(Rif)

Riferimenti

Numero di archivi, ATTRIBUTI, Nome campo, Nome archivio

Nome archivio — File name

Sintassi

Nome archivio(NumArchivio)
➤ Stringa

o

Nome archivio(Puntatore_Archivio)
➤ Stringa

Parametri	Tipo	Descrizione
NumArchivio	Espr_NumPos	Numero interno dell'archivio
Puntatore_Archivio	Punt_Archivio	Puntatore ad un archivio

Descrizione

Nome archivio restituisce il nome dell'archivio chiamato con il suo numero (ordine di creazione degli archivi della base dati) o con un puntatore a questo archivio.

Esempio

Effettueremo il caricamento in un vettore dei nomi di tutti gli archivi di una base dati.

```
VETTORE DI TESTI(Nomi;Numero di archivi)      Dichiara il vettore Nomi
Ciclo(i;1;Numero di archivi)                  Per ogni archivio
   Nomi{i}:=Nome archivio{i}                  Copia nel vettore Nomi il nome dell'archivio
Fine ciclo
```

Riferimenti

Nome campo, Numero di campi, Numero di archivi, ATTRIBUTI CAMPO

Nome campo

Field name

Sintassi

Nome campo(NumArchivio;NumCampo)
➤ Stringa

o

Nome campo(Puntatore_Campo)
➤ Stringa

Parametri	**Tipo**	**Descrizione**
NumArchivio	Espr_NumPos	Numero interno dell'archivio
NumCampo	Espr_NumPos	Numero di campo
Puntatore_Campo	Punt_Campo	Puntatore ad un campo

Descrizione

Nome campo restituisce una stringa di caratteri uguale al nome del campo chiamato con uno dei due modi seguenti:
- con il proprio numero di archivio e numero di campo
- con un puntatore a questo campo

Esempio

Eseguiremo il caricamento in un vettore dei nomi dei campi dell'Archivio 1

```
VETTORE DI TESTI(Nomi;Numero di campi(1))     Dichiara il vettore Nomi
Ciclo(i;1;Numero di Campi(1))                 Per ogni campo dell'Archivio 1
    Nomi{i}:=Nome campo(1;i)                  Copia il nome del campo nel vettore
Fine ciclo
```

Riferimenti

Nome archivio, Numero di campi, Numero di archivi, ATTRIBUTI CAMPO

Archivio

File

Sintassi

Archivio(NumArchivio)
➤ Puntatore_Archivio

o

Archivio(Puntatore_Archivio)
➤ NumArchivio

Parametri	**Tipo**	**Descrizione**
NumArchivio	Espr_NumPos	Numero interno dell'archivio
Puntatore_Archivio	Punt_Archivio	Puntatore ad un archivio

Descrizione
Archivio ammette due sintassi simmetriche. Se questa funzione riceve come argomento un numero di archivio, restituirà un puntatore a questo archivio. Se invece riceve un puntatore ad un archivio, restituirà il numero interno di archivio corrispondente.

Esempio
Eseguiremo il caricamento di un vettore di puntatori agli archivi della base dati.

VETTORE DI PUNTATORI(Rif;**Numero di archivi**)	Dichiara il vettore di puntatori
Ciclo(i;1;**Numero di archivi**)	Per tutti gli archivi
Rif{i}:=**Archivio**(i)	Copia il puntatore all'i-esimo archivio nell'elemento Rif{i}.
Fine ciclo	

Riferimenti
Campo

Campo — Field

Sintassi
Campo(NumArchivio;NumCampo)
➤ Puntatore_Campo

o

Campo(Puntatore_Campo)
➤ NumCampo

Parametri	**Tipo**	**Descrizione**
NumArchivio	Espr_NumPos	Numero interno dell'archivio
NumCampo	Espr_NumPos	Numero di campo
Puntatore_Campo	Punt_Campo	Puntatore ad un campo

Descrizione
Campo ammette due sintassi. Se questa funzione riceve come argomento un numero di archivio ed un numero di campo, restituirà un puntatore a questo campo. Se invece riceve un puntatore ad un campo, restituirà il numero di campo corrispondente.

Esempio
Eseguiremo il caricamento di un vettore di puntatori ai campi del primo archivio

VETTORE DI PUNTATORI(Rif;**Numero di campi**)	Dichiara il vettore di puntatori
Ciclo(i;1;**Numero di campi**)	Per tutti i campi dell'Archivio 1
Rif{i}:=**Campo**(1;i)	Copia un puntatore all'i-esimo campo nell'elemento Rif{i}.
Fine ciclo	

Riferimenti
Archivio

ATTRIBUTI CAMPO — FIELD ATTRIBUTES

Sintassi
ATTRIBUTI CAMPO(NumArchivio;NumCampo;Tipo;Lunghezza;Indicizzato)

o

ATTRIBUTI CAMPO(Puntatore_Campo;Tipo;Lunghezza;Indicizzato)

Parametri	Tipo	Descrizione
NumArchivio	Espr_NumPos	Numero interno dell'archivio
NumCampo	Espr_NumPos	Numero di campo
Puntatore_Campo	Punt_Campo	Puntatore ad un campo
Tipo	Espr_Var	Intero(0..11) corrispondente al tipo del campo
Lunghezza	Espr_Var	Intero contenente la lunghezza del campo
Indicizzato	Espr_Var	Booleano vero se il campo è indicizzato

Descrizione
ATTRIBUTI CAMPO restituisce nelle tre variabili *Tipo*, *Lunghezza* e *Indicizzato* la descrizione del campo a cui si riferisce per mezzo del numero di archivio e del numero di campo, oppure per mezzo di un puntatore a questo campo.

In funzione del tipo di campo, Tipo assumerà i seguenti valori:

Alfa	0
Reale	1
Testo	2
Disegno	3
Data	4
Booleano	6
Sotto archivio	7
Intero	8
Intero Lungo	9
Ora	11

Esempio

ATTRIBUTI(1;1;vTipo;vLunghezza;vIndice)

Rif:=»[Archivio1]Nome
ATTRIBUTI(Rif;vTipo;vLunghezza;vIndice)

La procedura seguente carica nei 2 vettori a due dimensioni i nomi ed i tipi di campi della base dati. vtCampo è un vettore a due dimensioni contenente il nome dei campi. vtTipo è un vettore a due dimensioni contenente il tipo dei campi.

Questo primo vettore vi servirà per la conversione tra il numero del tipo ed una rappresentazione sotto forma di lettera del tipo di campo.

VETTORE DI STRINGHE(1;IITipo;11)	Dichiarazione di IITipo
IITipo{0}:="A"	
IITipo{1}:="R"	
IITipo{2}:="T"	
IITipo{3}:="P"	
IITipo{4}:="D"	
IITipo{6}:="B"	
IITipo{7}:="*"	
IITipo{8}:="I"	
IITipo{9}:="L"	
IITipo{11}:="O"	
VETTORE DI TESTI(vtCampo;**Numero di archivi**;1)	Dichiara il vettore a due dimensioni vtCampo
VETTORE DI TESTI(vtTipo;**Numero di archivi**;1)	Dichiara vtTipo come vtCampo
Ciclo(i;1;**Numero di archivi**)	Per ogni archivio
VETTORE DI TESTI(vtCampo{i};**Numero di campi**(i))	Dichiara la i-esima colonna di vtCampo
VETTORE DI TESTI(vtTipo{i};**Numero di campi**(i))	Dichiara la i-esima colonna di vtTipo
Ciclo(j;1;**Numero di campi**(i))	Per ogni campo
vtCampo{i}{j}:=**Nome campo**(i;j)	vtCampo{i}{j} assume il valore del nome del campo
ATTRIBUTI CAMPO(i;j;vT;vL;vI)	Esamina gli attributi di questo campo
vtTipo{i}{j}:=IITipo{vT}+(**Num**(Vt=0)***Stringa**(vL))	Inserisci il tipo del campo e la sua lunghezza, se si tratta di una stringa nell'elemento {i},{j} di vtTipo
Fine ciclo	
Fine ciclo	

Questi due vettori non verranno visualizzati direttamente sullo schermo, in quanto si tratta di vettori a due dimensioni. Per farli apparire è sufficiente chiamare una procedura denominata MostraArchivio, alla quale passerete il numero di archivio da visualizzare.

COPIA VETTORE(vtCampo{$1};ICampi)	Copia la colonna $1 di vtCampo nei campi
COPIA VETTORE(vtTipo{$1};ITipi)	Copia la colonna $1 di vtTipo nei tipi

Riferimenti

Numero di archivi, Numero di campi, Nome campo, Archivio, Campo

Nome utente — Current user

Sintassi
Nome utente
➤ Alfanumerico

Descrizione
Nome utente restituisce il nome dell'utente che sta lavorando sulla base dati.

Esempio
Ecco un esempio che verrà eseguito all'apertura della base dati. Verrà visualizzato un messaggio di benvenuto a seconda della persona che utilizza la base dati.

```
Caso fra                                                         Se nella base dati è entrato il
   :(Nome utente="Super_Utente")|(Nome utente="Amministratore")  Super_Utente o l'Amministratore
      AVVISO("Benvenuto Capo")                                   Visualizza una finestra di benvenuto
   altrimenti                                                    Se si tratta di un altro utente
      H:=Ora attuale                                             Verifica l'ora di arrivo
      Se(H<†9:30:00†)                                            Se l'ora è inferiore alle 9.30
         AVVISO("Benvenuto")                                     Augura il benvenuto all'utente
      altrimenti
         AVVISO("Sei ancora in ritardo")                         Altrimenti, avvertilo che è in ritardo
      Fine se
Fine caso
```

Riferimenti
CAMBIA PAROLA ORDINE, CAMBIA UTENTE

MODIFICA PRIVILEGI — EDIT ACCESS

Sintassi
MODIFICA PRIVILEGI

Descrizione
Questo comando visualizza il catalogo degli utenti disponibile in ambiente Progetto e permette ai proprietari di gruppi, all'amministratore ed al Super_Utente di modificare l'organizzazione dei gruppi. In questo modo, un proprietario di gruppi può aggiungere o togliere un utente dal proprio gruppo.

Riferimenti
Nome utente, CAMBIA UTENTE

CAMBIA PAROLA ORDINE — CHANGE PASSWORD

Sintassi
CAMBIA PAROLA ORDINE(ParolaOrdine)

Parametri	Tipo	Descrizione
ParolaOrdine	Espr_Alfa	Nuova parola d'ordine

Descrizione
Il comando CAMBIA PAROLA ORDINE permette ad un utente di cambiare la propria parola d'ordine. Questo nuovo nome gli verrà richiesto alla successiva apertura della base dati.

Esempio
Ecco una procedura che permette all'utente di cambiare la propria parola d'ordine. Per maggior sicurezza, chiederemo all'utente di inserirla due volte. Se le parole sono diverse, non gli sarà possibile cambiare la parola d'ordine.

```
vPass1:=Richiedi("Introduci la nuova parola d'ordine")                 Richiedi la parola d'ordine una prima volta
Se(OK=1)
   vPass2:=Richiedi("Introduci una seconda volta la parola d'ordine")  Chiedi conferma della parola d'ordine
   Se(OK=1)
      Se(vPass1=vPass2)                                                Se le due parole d'ordine sono uguali
         CAMBIA PAROLA ORDINE(vPass1)                                  Cambia la parola d'ordine corrente
      altrimenti                                                       Altrimenti
         AVVISO("Le due parole d'ordine sono diverse")                 Avverti che sono diverse
      Fine se
   Fine se
Fine se
```

Riferimenti
CAMBIA UTENTE, Nome utente

CAMBIA UTENTE — CHANGE ACCESS

Sintassi
CAMBIA UTENTE

Descrizione
La chiamata del comando CAMBIA UTENTE visualizza il catalogo degli utenti esattamente come è stato presentato all'apertura della base dati. Potete così cambiare l'utente strada facendo. Dovrete reinserire la parola d'ordine per questo utente.

Riferimenti
MODIFICA PRIVILEGI, CAMBIA PAROLA ORDINE, Nome utente

Interruzioni

CHIAMA SU ERRORE — ON ERROR CALL

Sintassi
CHIAMA SU ERRORE(LaProcedura)

Parametri	Tipo	Descrizione
LaProcedura	Espr_Alfa	Nome della procedura per la gestione degli errori

Descrizione
CHIAMA SU ERRORE permette di impostare una procedura di interruzione, il cui nome è specificato da *LaProcedura*. Tale procedura verrà eseguita ogni volta che si verifica un errore durante l'esecuzione di un'istruzione. Ciò vi permette di controllare, all'interno della vostra applicazione, i possibili errori di esecuzione. Potete ottenere il codice di errore servendovi della variabile di sistema **Error**. Al termine della procedura di interruzione, 4th DIMENSION ritorna alle istruzioni interrotte, a meno che abbiate chiamato INTERROMPI (descritto qui di seguito) nella procedura di interruzione. La lista dei codici di errore si trova nell'Appendice di questo manuale.

Esempio
La procedura GestioneErrori, in funzione dell'errore rilevato, visualizza un messaggio di spiegazione:

```
Caso fra
    :(Errore=1006)
    IlMessaggio:="Procedura interrotta dall'utente, Premi [A-capo] per continuare"
    :(Errore=51)
    IlMessaggio:="Il modulo di mailing non è installato, usa 4D Mover")
altrimenti
    IlMessaggio:="Un errore N° "+Stringa(Errore)+"ha interrotto l'operazione".
Fine caso
AVVISO(IlMessaggio)
```

Riferimenti
INTERROMPI, CHIAMA SU EVENTO

CHIAMA SU PORTA SERIALE — ON SERIAL PORT CALL

Sintassi
CHIAMA SU PORTA SERIALE(LaProcedura)

Parametri	Tipo	Descrizione
LaProcedura	Espr_Alfa	Procedura da chiamare nel caso di una comunicazione

Descrizione
CHIAMA SU PORTA SERIALE imposta la procedura di interruzione *LaProcedura*, che verrà chiamata ogni volta che l'ultima porta seriale selezionata con IMPOSTA COMUNICAZIONE riceve un carattere. Quando un carattere arriva sulla porta seriale, se è in corso una procedura, questa verrà interrotta e *LaProcedura* verrà chiamata. Al termine dell'esecuzione di *LaProcedura*, 4th DIMENSION ritornerà alle istruzioni precedentemente in corso. Se decidete di non eseguire operazioni con i caratteri ricevuti, non dimenticate comunque di liberare il buffer seriale chiamando RICEVI BUFFER. In caso contrario, 4th DIMENSION richiamerà la procedura di interruzione.
Per disinstallare la procedura di interruzione, passate una stringa di caratteri vuota come nome della procedura.

Esempio

CHIAMA SU PORTA SERIALE("IlBuffer")

Riferimenti
CHIAMA SU EVENTO, CHIAMA SU ERRORE, RICEVI BUFFER

CHIAMA SU EVENTO — ON EVENT CALL

Sintassi
CHIAMA SU EVENTO(LaProcedura)

Parametri	Tipo	Descrizione
LaProcedura	Espr_Alfa	Nome della procedura di gestione degli eventi

Descrizione
CHIAMA SU EVENTO vi permette di installare la procedura di interruzione *LaProcedura* che verrà chiamata ogni volta che provocate un evento premendo un tasto sulla tastiera o facendo clic.
La procedura di interruzione è attiva soltanto se è in corso una procedura di 4th DIMENSION.
LaProcedura non verrà chiamata se provocate un evento mentre avete sullo schermo la barra di menu principale, in quanto in quel momento nessuna procedura di 4th DIMENSION è attiva.
LaProcedura potrà scoprire l'evento che ne ha provocato la chiamata verificando le variabili di sistema *KeyCode*, *MouseDown* e *Modifiers*.

- *KeyCode* restituisce il codice ASCII del carattere corrispondente al tasto premuto sulla tastiera
- *MouseDown* restituisce 1 se avete fatto clic
- *Modifiers* permette di sapere se sono stati premuti i tasti Comando, Opzione, Maiuscole, Blocco Maiuscole, o Control.

KeyCode, MouseDown e Modifiers restituiscono valori significativi soltanto durante l'esecuzione della procedura di interruzione LaProcedura.

Esempio

IMPOSTA ARCHIVIO CORRENTE([Clienti])
TUTTE LE SCHEDE
CHIAMA SU EVENTO("GestioneEvento") — Installa la procedura di interruzione
MESSAGGIO("Premi Opzione-Q per interrompere l'operazione") — Segnala che occorre premere Opzione-Q per interrompere la procedura
Ripeti
...
Finché(StopEvt=1)|(**Fine selezione**)
CHIAMA SU EVENTO("") — Disinstalla la procedura di interruzione

La procedura seguente vi fornisce la decodificazione completa della variabile Modifiers ed aggiorna le variabili Comando, Maiuscole, Blocco Maiuscole, Opzione, Control.
Se premete Opzione-Q, 4th DIMENSION chiama la procedura *GestioneEvento*, la variabile StopEvt restituisce 1 e la procedura si interrompe.

```
$M:=Modifiers\256                      Calcola la variabile Comando
Comando:=Modulo$M%2
$M:=Modifiers\512                      Calcola la variabile Maiuscole
Maiuscole:=Modulo$M%2
$M:=Modifiers\1024                     Calcola la variabile Blocco Maiuscole
Blocco Maiuscole:=Modulo$M%2
$M:=Modifiers\2048                     Calcola la variabile Opzione
Opzione:=Modulo$M%2
$M:=Modifiers\4096                     Calcola la variabile Control
Control:=Modulo$M%2
Se((Opzione=1)&(KeyCode=Ascii("Q"))    Se è stato premuto Opzione-Q
    StopEvt:=1                         Interrompi la procedura
altrimenti                             Altrimenti
    StopEvt:=0                         Continua
Fine se
```

Riferimenti

CHIAMA SU ERRORE

Indefinita

Undefined

Sintassi
Indefinita(LaVariabile)
➤ Booleano

Parametri	Tipo	Descrizione
LaVariabile	Espr_Var	Variabile di 4th DIMENSION

Descrizione
Indefinita restituisce un valore booleano vero se *LaVariabile* è indefinita, vale a dire se LaVariabile non esiste.

Nota:
Questa funzione diventerà superflua con il compilatore, poiché restituirà sempre un valore falso.

Riferimenti
Variabile, Tipo

REGISTRA VARIABILI

SAVE VARIABLE

Sintassi
REGISTRA VARIABILI(NomeArch;Var1;{Var2...VarN})

Parametri	Tipo	Descrizione
NomeArch	Espr_Alfa	Nome del documento su disco
Var1...VarN	Espr_Var	Variabile da scrivere su disco

Descrizione
REGISTRA VARIABILI vi permette di registrare il blocco di variabili *Var1, Var2,..., VarN* nel documento Macintosh *NomeArch*. Se *NomeArch* è una stringa vuota, 4th DIMENSION visualizza la finestra standard per la creazione dei documenti. Dovrete perciò scegliere il nome del documento. La varialbile di sistema OK restituirà 1 se avete registrato il documento, il cui nome ritroverete nella variabile Document, altrimenti OK restituirà 0. Il formato utilizzato da REGISTRA VARIABILI è diverso da quello di INVIA VARIABILE.

Esempio e riferimenti
CARICA VARIABILI

CARICA VARIABILI — LOAD VARIABLE

Sintassi
CARICA VARIABILI(NomeDoc;Var1;{Var2...VarN})

Parametri	Tipo	Descrizione
NomeDoc	Espr_Alfa	Nome del documento su disco
Var1...VarN	Espr_Var	Variabile da scrivere su disco

Descrizione
CARICA VARIABILI vi permette di rileggere il blocco delle variabili *Var1, Var2,..., VarN*, dal documento Macintosh il cui nome è *NomeDoc.*
Se passate una stringa vuota in *NomeDoc*, 4th DIMENSION visualizza sullo schermo la finestra standard per l'apertura dei documenti e dovrete quindi scegliere il nome del documento. La variabile di sistema OK restituirà 1 se avete riletto il documento, il cui nome ritroverete nella variabile Document, altrimenti OK restituirà 0.

Esempio
Prendiamo un archivio [Fatture] che presenta un campo [Fatture]Numero, nel quale è registrato il numero della fattura. CARICA VARIABILI e REGISTRA VARIABILI vi permettono di gestire automaticamente la numerazione delle fatture. Indichiamo il campo [Fatture]Numero come **Non inseribile**.
Nelle procedure di formato del formato di inserimento dell'archivio [Fatture] scriveremo:

```
Se(Prima)                                          Prima della comparsa del formato sullo schermo
   Se(Numero=0)                                    Numero è uguale a 0: si tratta di aggiunta di schede
      ELIMINA VARIABILE(xNum)                      Cancella dalla memoria la variabile xNum
      CARICA VARIABILI("LaBase.NumFatt";xNum)      Cerca il numero di fattura successivo
      Se(Indefinita(xNum))                         Se la variabile è indefinita, il documento non esiste
         xNum:=1                                   Aggiunta della prima fattura. Imposta 1 in xNum
      altrimenti
         xNum:=xNum+1                              Altrimenti incrementa xNum di una unità
      Fine se
      Numero:=xNum                                 Copia il numero della fattura nel campo Numero
   Fine se
Fine se
Se(Durante)                                        Istruzioni di Durante

Fine se
Se(Dopo)                                           Se l'inserimento è confermato
   Se(Precedente(Numero)=0)                        Il valore precedente del campo numero è uguale a
                                                   0: si tratta di un'aggiunta di fatture.
      REGISTRA VARIABILI("LaBase.NumFatt";xNum)    Aggiorna il numero della fattura nel documento "La-
   Fine se                                         Base.NumFatt" registrato.
Fine se
```

Riferimenti
REGISTRA VARIABILI

Note relative a CARICA VARIABILI e REGISTRA VARIABILI
Se volete esportare delle variabili in un documento destinato allo scambio di dati con un altro Macintosh, è preferibile usare INVIA PACCHETTO e RICEVI PACCHETTO.
Dopo l'esecuzione di CARICA VARIABILI, se fate un test e la variabile rimane indefinita o se presenta un valore invariato, significa che il documento non esiste. Occorre naturalmente cancellare le variabili in questione dalla memoria centrale prima di chiamare CARICA VARIABILI, usando ELIMINA VARIABILE (descritto qui di seguito), se volete essere sicuri della validità del test.

ELIMINA VARIABILE — CLEAR VARIABLE

Sintassi
ELIMINA VARIABILE(LaVar)

Parametri	Tipo	Descrizione
LaVar	Espr_Alfa o	Inizio del nome della/e variabile/i da cancellare
	Espr_Var	Variabile da cancellare

Descrizione
Se *LaVar* è un'espressione alfanumerica, ELIMINA VARIABILE cancella dalla memoria centrale tutte le variabili il cui nome inizia con *LaVar*.
Se *LaVar* è un'espressione variabile, ELIMINA VARIABILE cancella dalla memoria centrale *LaVar*.
Il contenuto delle variabili cancellate verrà irrimediabilmente perduto.
La prima sintassi di questo comando non ha alcun effetto se usate il compilatore di 4th DIMENSION.

Esempio

```
ELIMINA VARIABILE("v")          Cancella tutte le variabili il cui nome inizia con v
ELIMINA VARIABILE(vTotale)      Cancella la variabile vTotale, ma non vTotale2.
```

Puntatore a — Get pointer

Sintassi
Puntatore a(NomeVar)

Parametri	Tipo	Descrizione
NomeVar	Espr_Alfa	Nome della variabile

Descrizione
La funzione Puntatore a restituisce un puntatore alla variabile il cui nome è uguale a *NomeVar*. Se la variabile designata da *NomeVar* non esiste, verrà creata. Questa funzione, compatibile con il futuro compilatore di 4th DIMENSION, vi permette di effettuare delle indirezioni sui nomi delle variabili.

Esempio

```
Ciclo(i;1;10)                              Inizializza le variabili da MiaVar1 a MiaVar10
    $p:=Puntatore a("MiaVar"+Stringa(i))   $p è il puntatore alla mia variabile
    $p»:=""                                Inizializza la variabile puntata da $p
Fine ciclo
```

ESEGUI EXECUTE

Sintassi
ESEGUI(Istruzione)

Parametri	Tipo	Descrizione
Istruzione	Espr_Alfa	Istruzione che 4th DIMENSION deve eseguire

Descrizione
ESEGUI prende la stringa di caratteri *Istruzione*, verifica che la sintassi sia corretta, poi la esegue.
Se *Istruzione* è una stringa di caratteri vuota, ESEGUI non esegue alcuna azione.

Istruzione può contenere:
- la chiamata di un comando di 4th DIMENSION
- la chiamata di una procedura globale
- un assegnamento
- un riferimento a variabili globali

Istruzione non deve contenere:
- variabili locali
- istruzioni di controllo di flusso (Se, Caso fra, Mentre,...)

Se vi serve il carattere virgolette nella costruzione delle vostre istruzioni, inseritelo usando il comando Carattere(34).
Questo comando è stato mantenuto per mantenere la compatibilità con la versione 1.0.x di 4th DIMENSION.

ATTIVA MODO TRACCIA TRACE

Sintassi
ATTIVA MODO TRACCIA

Descrizione
ATTIVA MODO TRACCIA interrompe l'esecuzione della procedura in corso e visualizza la finestra Collaudo del modo Traccia. Per cancellare questa condizione fate clic sul pulsante No traccia nella finestra Collaudo, oppure chiamate nelle vostre procedure il comando DISATTIVA MODO TRACCIA (descritto qui di seguito).

Riferimenti
DISATTIVA MODO TRACCIA

DISATTIVA MODO TRACCIA — NO TRACE

Sintassi
DISATTIVA MODO TRACCIA

Descrizione
Dopo una chiamata di DISATTIVA MODO TRACCIA, 4th DIMENSION non visualizza più automaticamente la finestra del modo Traccia dopo ogni esecuzione dell'istruzione. Notate che la finestra del modo Traccia compare se in una delle vostre procedure avete incontrato l'istruzione ATTIVA MODO TRACCIA, oppure se dopo un messaggio di errore dell'interprete di 4th DIMENSION avete fatto clic sul pulsante Traccia.

Esempio
Realizzeremo una procedura *GestioneErrore* che cancellerà il passaggio al modo Traccia se interrompete l'esecuzione di una procedura.

```
Se(Errore=1006)                          Se l'errore è causato da una interruzione
   DISATTIVA MODO TRACCIA                Elimina il modo Traccia
Fine se

CHIAMA SU ERRORE("GestioneErrori")       Installa la procedura GestioneErrori
```

Riferimenti
ATTIVA MODO TRACCIA

Numero di parametri — Count parameters

Sintassi
Numero di parametri
➤ Intero positivo

Descrizione
Numero di parametri restituisce un intero corrispondente al numero di argomenti che una procedura di 4th DIMENSION riceve quando viene chiamata da un'altra procedura.

Esempio
Funzione MassimoFra

```
$0:=0                               All'avvio, il massimo vale 0: inizializza $0 a 0
Ciclo(i;1,Numero di parametri)      Effettua un Ciclo da 1 fino al numero dei parametri
   Se(${i}>$0)                      Se il parametro ${i} è superiore a $0
      $0:=${i}                      $0 assume il valore di ${i}
   Fine se
Fine ciclo

llMax:=MassimoFra(10;20;5;2;150;120)     Restituirà 150
```

ESCI DA 4D — QUIT 4D

Sintassi
ESCI DA 4D

Descrizione
ESCI DA 4D esce dalla base dati che state usando, chiude tutti gli archivi aperti da 4th DIMENSION e ritorna al Finder.

Esempio
Script del pulsante Ciao:

```
                                  Se viene fatto clic sul pulsante Ciao
ESCI DA 4D                        Esci dal programma
```

INTERROMPI — ABORT

Sintassi
INTERROMPI

Descrizione
Dovete chiamare questo comando unicamente all'interno di una procedura di interruzione sugli errori, che avete installato servendovi del comando CHIAMA SU ERRORE. Se nel corso di questa procedura chiamate INTERROMPI, 4th DIMENSION non ritornerà alle procedure interrotte. La chiamata del comando INTERROMPI equivale a fare clic sul pulsante Termina della finestra d'errore o sul pulsante Termina della finestra Collaudo nel modo Traccia.
Questo comando non avrà alcun effetto se usate il compilatore di 4th DIMENSION.

Riferimenti
CHIAMA SU ERRORE, CHIAMA SU EVENTO

REGISTRA TAMPONE — FLUSH BUFFERS

Sintassi
REGISTRA TAMPONE

Descrizione
Il comando REGISTRA TAMPONE provoca la scrittura su disco dei dati registrati temporaneamente nella memoria tampone di 4th DIMENSION. Questa funzione viene già gestita automaticamente da 4th DIMENSION, che libera una parte del tampone ogni volta che è necessario.
Questo comando è utile soltanto lavorando in mono-utenza.

I puntatori

I puntatori

4th DIMENSION vi mette a disposizione uno strumento di sintassi di tipo speciale, il puntatore.
L'uso dei puntatori vi consentirà di ottenere un ulteriore livello di generalizzazione nella scrittura delle vostre procedure, e di conseguenza una loro maggiore versatilità e trasportabilità.

Un puntatore permette di realizzare delle indirezioni su 4 tipi di oggetti: le variabili semplici, le variabili di tipo vettore, i campi e gli archivi, ossia permette di gestire una variabile, un campo, un archivio o un vettore non solo direttamente, ma attraverso un altro oggetto, e precisamente un puntatore. Ad esempio, un puntatore permette di effettuare operazioni su un campo, senza sapere necessariamente di quale campo si tratta.

Nota relativa al concetto di indirezione
Si può pensare di identificare un Macintosh con il suo numero di serie, esattamente come si identifica un archivio con il proprio nome, ma questo non è l'unico modo. Se il Macintosh è di Paolo, non identificheremo il Macintosh direttamente, ma per mezzo di Paolo.
Questo tipo di identificazione presenta un vantaggio: Paolo può cambiare Macintosh. Può essere un Macintosh Plus, un SE, un Mac II, o qualsiasi altro tipo di Macintosh, ma non importa, perchè vi riferirete sempre al Macintosh di Paolo (che è comunque solo ed unico).
Ciò che occorre rilevare da questo esempio, non è tanto la relazione di proprietà o di appartenenza tra Paolo ed il suo Macintosh, quanto il fatto che tale Macintosh viene designato per mezzo di Paolo, e che si passa da Paolo per arrivare al Macintosh.
Si parla quindi di indirezione quando l'oggetto designato non viene indicato con il riferimento che gli è proprio, ma per mezzo di un intermediario che lo identifica in modo univoco.

Un puntatore di 4th DIMENSION viene indicato con il carattere » (ottenuto con Opzione-Maiuscole-&), anteposto all'oggetto sul quale si vuole definire il puntatore.
La definizione di un puntatore ad un oggetto x verrà scritta nel modo seguente: xPunt:=»x.
L'utilizzo del valore puntato si indica con il simbolo » posto dopo il nome della variabile puntatore.

Sulla base dell'esempio precedente, potremmo cambiare il valore di x scrivendo: xPunt»:=10.
Ciò significa che l'oggetto puntato da xPunt, ad esempio x, assumerà il valore di 10.
Esamineremo ora questi quattro tipi di puntatore, uno per volta.

Nota per i programmatori che fanno uso dei linguaggi tradizionali
La nozione di puntatore implementata in 4th DIMENSION differisce da quella utilizzata in Pascal, C, o altri linguaggi di questo tipo. Un puntatore Pascal restituisce l'indirizzo di un oggetto in memoria, indipendentemente dall'oggetto puntato. Un puntatore di 4th DIMENSION non corrisponde ad un indirizzo in memoria, ma permette di realizzare delle indirezioni su variabili, campi, archivi e vettori.

Puntatore ad una variabile semplice

Un puntatore ad una variabile si indica in questo modo: *vPunt*:=»MiaVariabile. Una volta definito questo puntatore, indipendentemente dal valore di MiaVariabile, vPunt punterà a questa variabile.
Potremmo scrivere, ad esempio:
AVVISO(vPunt»)
che sarà equivalente a:
AVVISO(MioValore)

Esempio
Immaginiamo di realizzare una procedura che abilita o disabilita un pulsante in base al valore di un'espressione booleana. Potremmo scrivere una procedura AbilitaPulsante.

```
Se(Condizione)
   ABILITA PULSANTE(IlPulsante)
altrimenti
   DISABILITA PULSANTE(IlPulsante)
Fine se
```

Se vi fermate qui, la procedura dovrà essere riscritta per ogni pulsante. Potete usare un puntatore per rendere la procedura generica. In effetti, un puntatore punta ad un oggetto senza che questo debba essere predefinito. Possiamo quindi riscrivere la procedura nel modo seguente:

```
                                  $1 = Espr_Bool e $2 = puntatore al pulsante da abilitare o disabilitare
Se($1)                            Se la condizione è vera
   ABILITA PULSANTE($2»)          Abilita il pulsante puntato da $2
altrimenti                        Se la condizione è falsa
   DISABILITA PULSANTE($2»)       Disabilita il pulsante puntato da $2.
Fine se
```

La chiamata di questa procedura si ottiene come segue:
AbilitaPulsante (Condizione1;»Pulsante1)
AbilitaPulsante (Condizione2;»Pulsante2)

Puntatore ad un archivio

Potete definire una variabile di tipo puntatore ad archivio nel modo seguente:
vPunt:=»[Archivio]
Avete a disposizione anche la funzione **Archivio**, che restituisce il puntatore all'archivio corrispondente al numero passato. Scriviamo una procedura generica di aggiunta schede, *MiaAggiunta*.

```
IMPOSTA ARCHIVIO CORRENTE($1»)              $1 = puntatore all'archivio
Ripeti
   AGGIUNGI SCHEDA
Finché(OK=0)
```

La chiamata di questa procedura si ottiene nel modo seguente:
MiaAggiunta(»[Archivio1])
o
MiaAggiunta(**Archivio**(x))
dove x è il numero dell'archivio che volete raggiungere.

Puntatore ad un campo

Potete anche definire una variabile di tipo puntatore a campo, nel modo seguente:
vPunt:=»[Archivio1]Campo
restituisce un puntatore a Campo di [Archivio1]
Potete anche usare la funzione **Campo**, che restituisce il puntatore al campo.
vPunt:=**Campo**(1;1)
restituisce un puntatore al primo campo del primo archivio.

Immaginiamo che vogliate accedere ad un campo f dell'archivio [Dati] per mezzo del numero corrispondente.

Userete dei puntatori ad un campo.
Se prendiamo $1 come numero del campo f ed $2 come valore da inserire, si può scrivere una nuova procedura *Assegna*:

v:=**Campo**(1;$1+1)
v»:=$2

La chiamata alla procedura *Assegna* si ottiene in questo modo:

Aggiunta(1;"Buongiorno") *Aggiunta*(2;"Signora")	La prima riga inserisce "Buongiorno" nel campo f1 e la seconda inserisce "Signora" nel campo f2.

Se usate frequentemente questo tipo di operazione, può esservi utile creare un vettore di puntatori ai campi dell'archivio con cui state lavorando. Questo vettore verrà caricato una volta per tutte nella procedura *Startup*. Effettuate il caricamento di un vettore di puntatori ad un campo per l'Archivio 1.

```
VETTORE DI PUNTATORI(vPunt;Numero di campi(1))
Ciclo(NumCampo;2;Numero di campi(1))
    vPunt{NumCampo}:=Campo(1;NumCampo))
Fine ciclo
```

L'aggiornamento si ottiene con la procedura chiamata *Assegnabis*, dove $1 è il numero del campo f e $2 il valore da inserire.

```
vPunt{$1}»:=$2
```

Nota per la versione 3/versione 4
Nella Versione 1.0.x di 4th DIMENSION, avreste utilizzato il comando **ESEGUI** per scrivere una procedura *Assegna* del tipo seguente:

```
ESEGUI("[Dati]f"+Stringa($1)+":="+$2)          $1 = numero del campo f e $2 = valore da inserire
```

Questo metodo presenta due inconvenienti:
- è lento nell'esecuzione
- se volete cambiare il nome dell'archivio o quello dei campi, dovrete riscrivere l'istruzione.

Puntatore ad un vettore

Potete definire allo stesso modo i puntatori ai vettori. Ci occuperemo di queste variabili a parte, in quanto si tratta di un tipo un po' speciale. Non è possibile passare un vettore come parametro di una procedura di 4th DIMENSION. Potete invece passare un puntatore a questo vettore. La procedura seguente *MiaAggiunta* aggiunge nel vettore puntato da $1 il valore ricevuto come secondo argomento.
$1 è un puntatore al vettore e $2 il valore da aggiungere. Scrivete :

```
INSERISCI ELEMENTI IN VETTORE($1»;Dimensione Vettore($1»)+1)
$1»{Dimensione Vettore($1»)}:=$2
```

L'aggiunta si ottiene scrivendo:
MiaAggiunta(»MioVettore;MioValore)
MiaAggiunta(»MioAltroVettore;MioValore)

Puntatore ad un puntatore

E' possibile definire un puntatore ad un altro puntatore. Siete nella seguente situazione: a punta verso b che punta verso c. L'accesso a c partendo da a si ottiene tramite una variabile intermedia. Scrivete:
$v:=a»
$v2:=$v»
$v2 conterrà il valore di c.

Non potreste scrivere
$v2:=a»»
Questa sintassi è errata.

Direttive di compilazione

Osservazioni generali

Questa parte si occupa delle regole di scrittura che dovrete rispettare usando il futuro compilatore di 4th DIMENSION. Questo prodotto, venduto separatamente, vi permetterà di accelerare l'esecuzione delle procedure, con un fattore che varierà da 3 a 1000.
Per poter sfruttare tutte le possibilità del compilatore, occorre che talvolta vi dobbiate attenere ad uno stile di programmazione che non vi è imposto da 4th DIMENSION in versione interpretata. Le direttive di compilazione avranno senso solo se vi atterrete alle regole seguenti.

Ridefinizione dei tipi delle variabili

Un compilatore collega tutte le procedure, analizza tutte le variabili, ne deduce il tipo, e prepara una tabella di variabili. Ciò presuppone che il tipo della variabile sia noto prima dell'esecuzione delle procedure. Se il compilatore rileva che una delle variabili cambia il proprio tipo nel corso dell'elaborazione, la compilazione verrà interrotta e dovrete correggere le procedure.
Vi consigliamo quindi di prendere nota dei nomi e del tipo di variabili che usate, per evitare sorprese in seguito. Scrivete, ad esempio:

```
x:=Richiedi("Numero del mese")
Se(OK=1)
    x:=Num(x)
    CERCA([Fatture]Mese=x)
Fine se
```

In un caso x è alfanumerico, nell'altro è numerico.
4th DIMENSION permette per la stessa variabile un cambiamento di tipo ed interpreta questo cambiamento in funzione del contesto ogni volta che compare la variabile.
Il compilatore deve avere una visione globale della procedura da compilare per poter effettuare la compilazione.
Una variabile con lo stesso nome non può essere alfanumerica in un punto e numerica in un altro.
Occorre avere due variabili con nomi diversi.
Quindi, per compilare la procedura suddetta, ovrete cambiare il nome di tutte le occorrenze della variabile x in uno dei due tipi.

Indirezione sui nomi delle variabili

Un compilatore non conserva i nomi delle variabili, ma soltanto il riferimento ad esse. Il compilatore non potrà quindi compilare i programmi usando delle indirezioni sui nomi delle variabili. Per accedere ad un gruppo di tre radio bottoni, r1, r2, r3, scrivete:

```
Se(Durante)
    Campo:=r1+(r2*2)+(r3*3)
Fine se
Se(Prima)
    r{Campo}:=1
Fine se
```

Il ciclo di **Prima** presenta una indirezione sul nome della variabile r.
Questa operazione non può essere compilata. Se volete che la procedura sia compilabile, avete a disposizione due soluzioni:
- usare **Caso fra**
- creare una funzione di accesso a questi pulsanti

Utilizzando Caso fra:

```
Caso fra
    :(Campo=1)
    r1:=1
    (Campo=2)
    r2:=1
    :(Campo=3)
    r3:=1
Fine caso
```

Questa procedura verrà eseguita senza problemi da parte del compilatore.
Potete ovviamente scegliere una gestione maggiormente ottimizzata dei pulsanti, ed in particolare creare una procedura più generale.
Potete creare una funzione che restituisca un puntatore ad un pulsante. Potrete in seguito gestire questi pulsanti tramite il puntatore. Creiamo una funzione *PrendiPulsante*.
$1 rappresenta il numero del pulsante.

```
Caso fra
    :($1=1)
    $0:=»r1
    :($1=2)
    $0:=»r2
    :($1=3)
    $0:=»r3
Fine caso
```

La chiamata nel ciclo di **Prima** si ottiene in questo modo:

```
Se(Prima)
    PrendiPulsante(Campo)»:=1
Fine se
```

Se dovete accedere frequentemente ad un elenco di variabili qualsiasi in un formato, vi consigliamo di creare un vettore di puntatori alle variabili, e di gestire il tutto partendo dal vettore di puntatori, dato che questo metodo è più rapido.

Indefinita — Undefined

La funzione **Indefinita** non ha senso in un programma compilato, in quanto, come abbiamo appena visto, un compilatore esamina le variabili in modo definitivo. Questa funzione restituirà quindi sempre Falso. Non dovete perciò basarvi sulla funzione **Indefinita** per sapere se dovete inizializzare una variabile o un campo qualsiasi.

ELIMINA VARIABILE — CLEAR VARIABLE

Il comando ELIMINA VARIABILE ha lo scopo di cancellare dalla memoria la o le variabili passate come argomento. Un compilatore prepara tutte le variabili in modo definitivo e statico.
Non potete quindi distruggere una variabile.
Come avete visto nella descrizione di questo comando, ELIMINA VARIABILE accetta due sintassi.

ELIMINA VARIABILE("a")
Questa prima sintassi cancella tutte le variabili il cui nome inizia con "a". In un programma compilato, questo comando verrà ignorato, poichè si tratta di una indirezione sul nome di una variabile.

ELIMINA VARIABILE(a)
Questa seconda sintassi cancella la variabile a. In un programma compilato, questo comando non distruggerà la variabile, ma la inizializzerà ad un valore nullo.

Nota per la Versione 1.0.x / Versione 4.x.x
Nella versione 1.0.x di 4th DIMENSION, potevate realizzare dei vettori partendo dall'indirezione sulle variabili (rappresentata dal simbolo §). Questi "vettori" non saranno compilabili.
Questa funzione è stata sostituita nella nuova versione da una gestione più efficace dei vettori.

I comandi indicati qui di seguito sono utili soltanto nel caso in cui vogliate utilizzare il compilatore di 4th DIMENSION.
Non hanno alcun effetto sul funzionamento del programma se questo non è compilato.

C_INTERO

C_INTEGER

Sintassi
C_INTERO(Var1;{....;VarN})

Parametri	**Tipo**	**Descrizione**
Var1...VarN	Espr_Var	Variabili da dichiarare come Intero

Descrizione
Il comando C_INTERO permette di dichiarare esplicitamente il tipo delle variabili *Var1...VarN* come intero. Sulla versione attuale del compilatore, le variabili dichiarate come Intero sono in effetti usate come Intero Lungo. Questo comando è disponibile per le versioni del compilatore che gestiscono gli Interi.

Riferimenti
C_INTERO LUNGO

C_INTERO LUNGO

C_LONGINT

Sintassi
C_INTERO LUNGO(Var1;{....;VarN})

Parametri	**Tipo**	**Descrizione**
Var1...VarN	Espr_Var	Variabili da dichiarare come Intero Lungo

Descrizione
C_INTERO LUNGO permette di dichiarare che le variabili *Var1..VarN* sono di tipo Intero Lungo. Questo comando presenta il doppio vantaggio di permettere al compilatore di eliminare eventuali ambiguità nella definizione dei tipi e di accelerare notevolmente le operazioni numeriche su questa variabile. Dovete chiamare questo comando soltanto se siete sicuri che i valori assunti da *Var1..VarN* sono strettamente compatibili con il tipo Intero Lungo (valore intero compreso tra ±2 miliardi).

Riferimenti
C_INTERO

C_REALE

C_REAL

Sintassi
C_REALE(Var1;{....;VarN})

Parametri	**Tipo**	**Descrizione**
Var1..VarN	Espr_Var	Variabili da dichiarare come Reale

Descrizione
C_REALE vi permette di dichiarare esplicitamente che le variabili *Var1..VarN* sono di tipo Reale. Questa funzione è utile soltanto nel caso in cui il compilatore non possa dedurre il tipo delle variabili *Var1..VarN*.

Riferimenti
C_INTERO LUNGO, C_INTERO

C_STRINGA — C_STRING

Sintassi
C_STRINGA(Lunghezza;Var1;{...;VarN})

Parametri	Tipo	Descrizione
Lunghezza	Espr_Num[2..80]	Lunghezza della stringa alfanumerica
Var1..VarN	Espr_Var	Variabile da dichiarare come stringa di lunghezza fissa

Descrizione
C_STRINGA permette di dichiarare esplicitamente che le variabili *Var1..VarN* sono stringhe la cui lunghezza massima è uguale a *Lunghezza*. Questa funzione consente al compilatore di impegnare meno spazio in memoria e di ottimizzare notevolmente la gestione delle variabili *Var1..VarN*.

Riferimenti
C_TESTO

C_TESTO — C_TEXT

Sintassi
C_TESTO(Var1;{...;VarN})

Parametri	Tipo	Descrizione
Var1..VarN	Espr_Var	Variabili da dichiarare come Testo

Descrizione
C_TESTO permette di dichiarare esplicitamente che le variabili *Var1..VarN* sono variabili di tipo Testo. Questo comando è utile soltanto per eliminare eventuali ambiguità nella compilazione del programma.

Riferimenti
C_STRINGA

C_IMMAGINE — C_PICTURE

Sintassi
C_IMMAGINE(Var1;{...;VarN})

Parametri	Tipo	Descrizione
Var1..VarN	Espr_Var	Variabili da dichiarare come Immagine

Descrizione
C_IMMAGINE permette di dichiarare esplicitamente che le variabili *Var1..VarN* sono di tipo Immagine. Questo comando è utile soltanto per eliminare eventuali ambiguità durante la compilazione del programma.

C_DATA — C_DATE

Sintassi
C_DATA(Var1;{...;VarN})

Parametri	Tipo	Descrizione
Var1..VarN	Espr_Var	Variabili da dichiarare come Data

Descrizione
C_DATA permette di dichiarare esplicitamente che le variabili *Var1..VarN* sono di tipo Data. Questo comando è utile soltanto per eliminare eventuali ambiguità durante la compilazione del programma.

C_PUNTATORE — C_POINTER

Sintassi
C_PUNTATORE(Var1;{...;VarN})

Parametri	Tipo	Descrizione
Var1..VarN	Espr_Var	Variabili da dichiarare come Puntatore

Descrizione
C_PUNTATORE permette di dichiarare esplicitamente che le variabili *Var1..VarN* sono di tipo Puntatore. Questo comando è utile soltanto per eliminare eventuali ambiguità durante la compilazione del programma.

C_BOOLEANO — C_BOOLEAN

Sintassi
C_BOOLEANO(Var1;{...;VarN})

Parametri	**Tipo**	**Descrizione**
Var1..VarN	Espr_Var	Variabili da dichiarare come Booleano

Descrizione
C_BOOLEANO permette di dichiarare esplicitamente che le variabili *Var1..VaarN* sono di tipo Booleano. Questo comando è utile soltanto per eliminare eventuali ambiguità durante la compilazione del programma.

C_ORA — C_TIME

Sintassi
C_ORA(Var1;{...;VarN})

Parametri	**Tipo**	**Descrizione**
Var1..VarN	Espr_Var	Variabili da dichiarare come Ora

Descrizione
C_ORA permette di dichiarare esplicitamente che le variabili *Var1..VarN* sono di tipo Ora. Questo comando è utile soltanto per eliminare eventuali ambiguità durante la compilazione del programma.

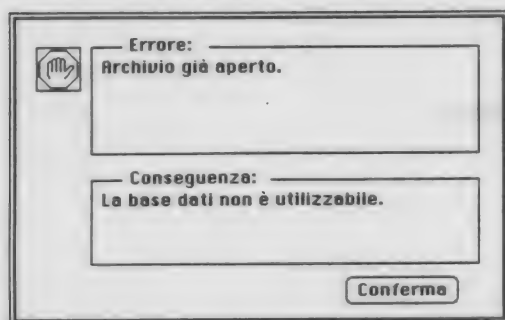
Multi-utenza: nozioni

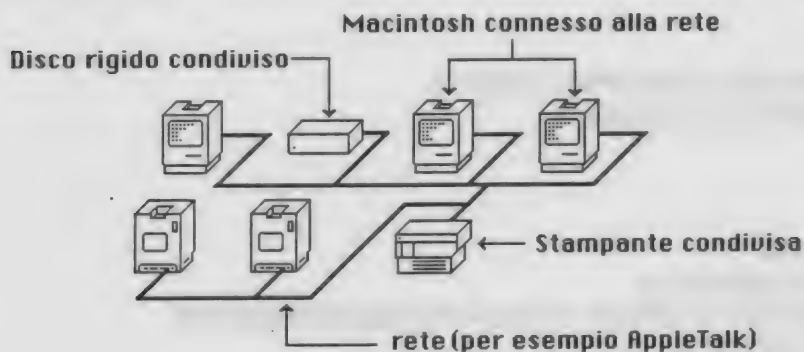
Struttura del capitolo

1. **Introduzione**
   1. Funzionamento in mono-utenza
   2. Funzionamento in multi-utenza
2. **Avvio in multi-utenza**
   1. Scelta del tipo di funzionamento
   2. Apertura di una base dati in multi-utenza
   3. Ubicazione degli archivi di una base dati gestita in multi-utenza
   4. La memoria tampone del Macintosh
3. **Particolarità della multi-utenza**
   1. Accesso alla struttura
   2. Selezioni e schede correnti
   3. Consultazioni, aggiunte di schede
   4. Modifiche, aggiornamenti e cancellazioni di schede
   5. Tipo di accesso alle schede
   6. I semafori
   7. Le transazioni
4. **I comandi di 4th DIMENSION e la multi-utenza**
   1. I comandi che caricano una scheda in memoria
   2. I comandi che caricano una scheda in memoria senza lo scopo esplicito di riscriverla
   3. I comandi che ammettono la riscrittura di una scheda
      - 3.1 APPLICA ALLA SELEZIONE
      - 3.2 MODIFICA SCHEDA
      - 3.3 MODIFICA SELEZIONE
      - 3.4 RICEVI SCHEDA, REGISTRA RELAZIONE PRECEDENTE
      - 3.5 REGISTRA SCHEDA, REGISTRA RELAZIONE
      - 3.6 CANCELLA SCHEDA
      - 3.7 CANCELLA SELEZIONE
      - 3.8 VETTORE IN SELEZIONE
   4. I comandi specifici della multi-utenza
   5. I comandi che cambiano effetto in multi-utenza
      - 5.1 MODIFICA PRIVILEGI, REGISTRA TAMPONE, ORDINA PERMANENTEMENTE
      - 5.2 DISIMPILA SCHEDA, IMPILA SCHEDA
   6. I comandi che accedono ai documenti su disco
5. **I server di rete**

1. Introduzione

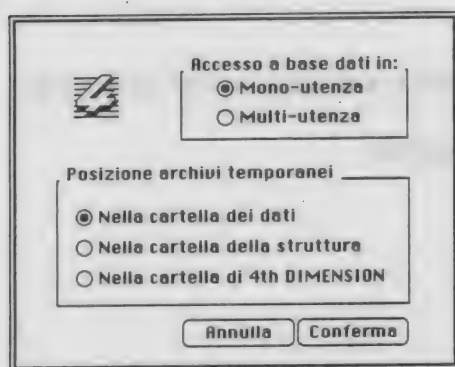
4th DIMENSION è un database relazionale che permette la gestione degli archivi sia in **mono-utenza** che in **multi-utenza**.

1. Funzionamento in mono-utenza
La configurazione di 4th DIMENSION è predisposta per la gestione in mono-utenza, che autorizza l'uso di una base dati da parte di un solo utente. Se due utenti aprono contemporaneamente la stessa base dati, il secondo di essi vedrà sullo schermo il seguente messaggio di errore:

Si tratta del tipo di gestione usato con più frequenza.

2. Funzionamento in multi-utenza
Permette a diversi utenti di lavorare contemporaneamente con la stessa base dati. Il funzionamento in multi-utenza presuppone un ambiente di lavoro provvisto di una rete (ad esempio **AppleTalk**), alla quale sono collegati i Macintosh degli utenti, ed un disco rigido comune, sul quale si trova la base dati destinata all'uso in multi-utenza:

2. Avvio in multi-utenza

Sia che utilizziate un Runtime o un ambiente di sviluppo di 4th DIMENSION, la stessa versione dell'applicazione vi permetterà entrambi i tipi di gestione.

1. Scelta del tipo di funzionamento

Per scegliere il tipo di gestione delle base dati di 4th DIMENSION avviate il programma a livello di **Finder**, tenendo premuto il bottone del mouse. Il programma visualizza la finestra di dialogo per la scelta del tipo di gestione prima di quella per l'apertura e la creazione della base dati.

Questa finestra di dialogo si compone di due parti:

- la parte superiore, **Accesso alla base dati in:**, permette di scegliere il tipo di gestione: fate clic sul radio bottone desiderato, **Mono-utenza** o **Multi-utenza**. La configurazione di 4th DIMENSION è predisposta per la gestione in **Mono-utenza**.

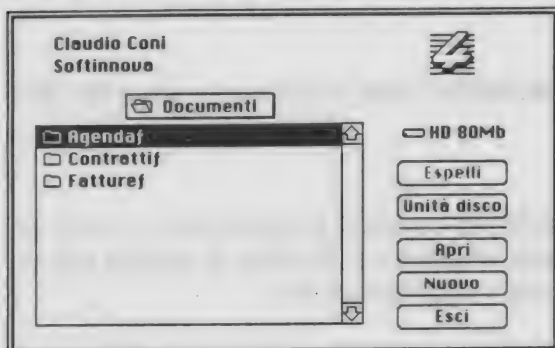
- la parte inferiore, **Posizione archivi temporanei**, permette di scegliere il luogo, su disco, in cui 4th DIMENSION creerà ed utilizzerà gli archivi temporanei durante l'uso della base dati. La configurazione di 4th DIMENSION è predisposta per inserire gli archivi temporanei nella cartella in cui si trova il documento contenente i dati della base dati, vale a dire quello il cui nome presenta il suffisso **.data**.

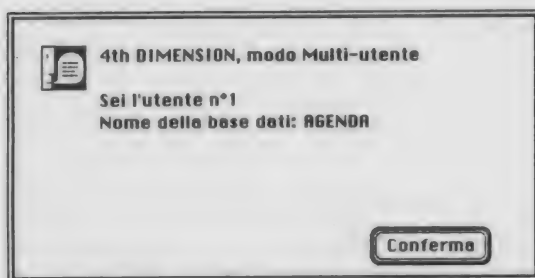
Una volta scelto il tipo di gestione, fate clic sul pulsante **Conferma** per confermare (oppure premete il tasto **A-capo** o **Inserimento**). Se vi accorgete di avere commesso un errore, fate clic sul pulsante **Annulla**.

Se confermate i parametri scelti in questa finestra, 4th DIMENSION li registrerà. Nelle fasi di lavoro successive, 4th DIMENSION riprenderà questi parametri senza che li dobbiate specificare ogni volta che aprirete una base dati. Per cambiare i parametri, è sufficiente visualizzare nuovamente la finestra per la scelta del tipo di gestione quando avviate il programma.

La scelta del tipo di gestione delle basi dati avviene nello stesso modo se usate la versione Runtime di 4th DIMENSION.

2. Apertura di una base dati in multi-utenza

Se confermate o annullate la finestra per la scelta del tipo di gestione, 4th DIMENSION visualizzerà la seguente finestra di dialogo:

Aprite o create la base dati che desiderate con il tipo di gestione scelto in precedenza. Per ritornare al **Finder**, fate clic sul pulsante **Esci**.

Quando aprite una base dati in multi-utenza, 4th DIMENSION ve lo segnala:

Questo messaggio visualizza il vostro numero di utente della base dati ed il nome della base dati aperta.

Quando una base dati di 4th DIMENSION è aperta in multi-utenza, il programma crea una nuova icona a livello di **Finder**, che rappresenta il documento che contiene i **semafori** della base dati.

Una base dati impiegata in multi-utenza si compone quindi di almeno tre documenti:

Il documento della **struttura** della base dati: contiene la descrizione degli archivi, dei campi, dei formati, delle procedure, e così via.

Il documento dei **dati** della base dati: contiene le schede, gli indici, e così via.

Il documento dei **semafori** della base dati: contiene le informazioni che permettono al programma di sapere se una determinata operazione può essere compiuta in quel momento.

Note

Per una corretta utilizzazione della base dati e per conservarne la coerenza, vi consigliamo di non cambiare i nomi di questi documenti, né la loro posizione sul disco mentre state usando la base dati.

Il nome del documento contenente i semafori è uguale a quello del documento dei dati della base dati, seguito dal carattere "_". Se, ad esempio, l'archivio dei dati si chiama ***Fatture 1989.data****, quello dei semafori si chiamerà* ***Fatture 1989.data_****.*
Il documento dei semafori viene creato nella cartella in cui si trova il documento dei dati.

3. Ubicazione degli archivi di una base dati gestita in multi-utenza

Quando una base dati è gestita in multi-utenza, i documenti dei dati e dei semafori vengono usati in comune dai diversi utenti. Questi due documenti devono perciò trovarsi in una stessa cartella condivisa sulla rete accessibile ad ogni utente.

L'ubicazione degli altri documenti, ossia il programma, il documento della struttura e i documenti temporanei, può variare a seconda dell'ambiente di lavoro di ogni utente, vale a dire in funzione della presenza o assenza di un disco rigido locale collegato al Macintosh di ogni utente:

- il **documento della struttura** della base dati può essere condiviso da tutti gli utenti; ciò significa che ogni utente può aprire la stessa icona situata in una cartella condivisa in rete. Ogni utente può anche avere una copia del documento della struttura situata sul disco rigido locale. Il vantaggio della seconda soluzione è l'aumento della velocità di esecuzione, nel senso che gli accessi al documento della struttura non avvengono attraverso la rete. In tal caso, occorre che tutti i documenti della struttura aperti dagli utenti che utilizzano la base dati creata siano identici.

- i **documenti temporanei della base dati** di ogni utente possono essere collocati in una cartella comune condivisa in rete o sul disco rigido locale di ogni utente. Come per il documento della struttura, la scelta di situare questi archivi su un disco rigido locale permette di aumentare la velocità di esecuzione, nel senso che gli accessi a questi documenti non avvengono attraverso la rete, ma localmente. Come abbiamo visto in precedenza, per mezzo della finestra di dialogo per scelta del tipo di gestione potete decidere l'ubicazione di questi documenti.

Note
I documenti temporanei creati durante l'uso di una base dati vengono usati in modo trasparente da 4th DIMENSION per registrare le selezioni sugli archivi e la relativa memoria tampone interna in cui vengono registrati i dati, qualora la memoria centrale del Macintosh si rivelasse insufficiente.
I documenti temporanei di selezione sono chiamati ***TempSel****, seguito da indicazioni che variano in base alla selezione e all'utente della base dati. Il numero di documenti* ***TempSel*** *varia in funzione delle selezioni create e della memoria disponibile.*
Il documento temporaneo della memoria tampone dei dati è chiamato ***TempCache****. Ne esiste uno per ogni utente. Nel caso della gestione in multi-utenza, al nome del documento viene aggiunta un'ulteriore indicazione per evitare eventuali conflittualità tra i vari documenti.*
Anche se intendete usare la base dati in mono-utenza, potete attivare la finestra di dialogo per la scelta del tipo di gestione per scegliere l'ubicazione dei documenti temporanei. Si tratta di un'operazione di grande utilità se il disco contenente la base dati è saturo; potrete così collocare i documenti temporanei su un altro disco.

La collocazione dei documenti temporanei su un disco rigido diverso da quello in cui si trova il documento dei dati presenta anche un altro vantaggio, indipendentemente dal tipo di gestione scelto per la base dati. Consente una maggiore velocità di esecuzione, nel senso che gli spostamenti delle testine del disco rigido avvengono in misura minore. In effetti, quando i documenti temporanei si trovano su un disco rigido diverso da quello del documento dei dati della base dati, la lettura o la scrittura in uno dei documenti temporanei non comporta lo spostamento delle testine del disco rigido in cui si trova il documento dei dati della base dati. Questo riparmio di tempo è ancor più rilevante nel caso di grosse operazioni su basi dati di grossa portata.

4. La memoria tampone del Macintosh

Quando lavorate su una base dati gestita in multi-utenza, assicuratevi che la memoria tampone del Macintosh sia completamente disattivata. A questo scopo, scegliete **Pannello di Controllo** nel menu Mela. Fate clic sull'icona **General** e sul radio bottone **Off** dell'area riservata alla **Memoria Tampone**:

Questa operazione è estremamente importante. Occorre infatti che i **dati letti e scritti** nel documento dei dati vengano anche **sempre letti e scritti sul disco rigido condiviso** dagli utenti. Se non rispettate la condizione di assenza di memoria tampone potreste causare la distruzione parziale dei dati della base dati.

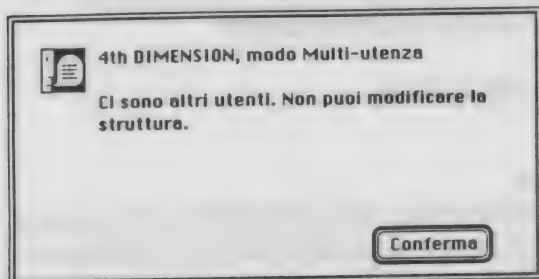
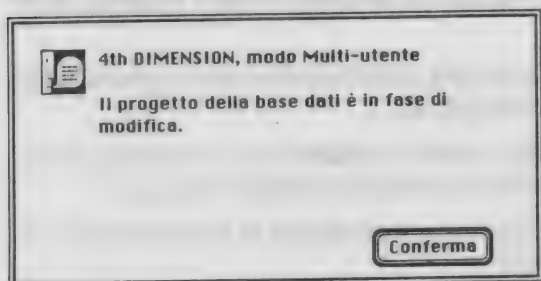
---

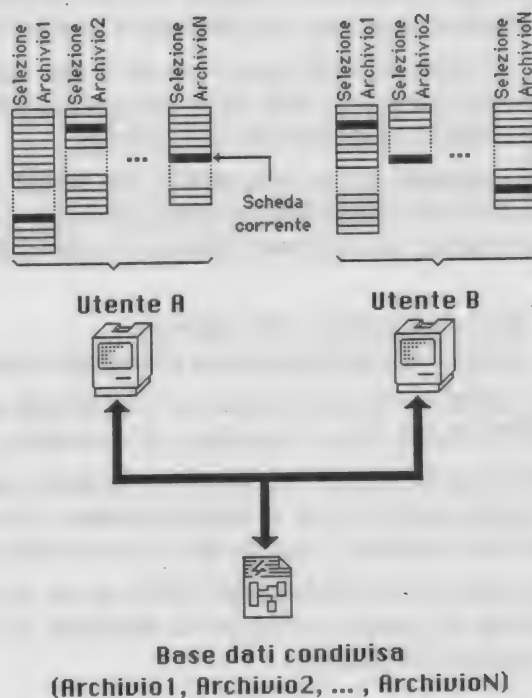
3. Particolarità della multi-utenza

Esamineremo ora i diversi impieghi di una base dati, a seconda del tipo di gestione scelto: mono-utenza o multi-utenza.

1. Accesso alla struttura

Il funzionamento in multi-utenza consente l'impiego di una base dati da parte di diversi utenti in ambiente **Utilizzo** o **Personalizzato**. L'accesso all'ambiente **Progetto** viene invece impedito quando un utente sta già usando la base dati. In effetti sarebbe molto pericoloso se un utente potesse modificare la struttura di una base dati mentre altri utenti stanno lavorando sui dati della stessa base dati.

Se un utente cerca di passare in ambiente **Progetto** mentre altri utenti stanno già lavorando sulla base dati, 4th DIMENSION visualizza il seguente messaggio:

Se una base dati è già aperta in ambiente **Progetto**, sullo schermo di ogni utente che cerca di aprirla verrà visualizzato il seguente messaggio:

2. Selezione e schede correnti

Quando una base dati è impiegata in multi-utenza, ogni utente dispone delle proprie selezioni e delle proprie schede correnti.
Non esiste infatti nessuna relazione di causa-effetto tra le diverse selezioni e schede correnti delle varie postazioni.
Ogni utente è libero di cercare le schede e selezionarne una come scheda corrente, indipendentemente dalle azioni degli altri.

3. Consultazioni, aggiunte di schede

Diversi utenti possono nello stesso momento:
- cercare, ordinare, stampare, creare grafici delle selezioni nello stesso archivio o in archivi diversi
- effettuare percorsi sequenziali sulle selezioni dello stesso archivio o in archivi diversi
- aggiungere schede nello stesso archivio o in archivi diversi
- usare le stesse schede contemporaneamente a scopo di consultazione.

4. Modifiche, aggiornamenti e cancellazioni di schede

Due utenti non possono modificare la stessa scheda nello stesso istante. Se la base dati autorizzasse la modifica contemporanea della scheda, il primo degli utenti che conferma le sue modifiche vedrebbe andare in fumo il proprio lavoro non appena un altro utente deciderà di confermare le proprie. Infatti, la seconda conferma della scheda cancellerebbe le modifiche apportate dal primo utente.

La base dati obbliga quindi il secondo utente ad attendere la conferma della scheda da parte del primo. Due utenti non possono modificare una scheda contemporaneamente: lo potranno fare soltanto **l'uno dopo l'altro**.

Questa limitazione non riguarda unicamente la modifica di una scheda, ma si riferisce più in generale alla **riscrittura di una scheda già esistente** (modifica o cancellazione).

Se chiedete di riscrivere una scheda quando invece vi potete accedere soltanto per la consultazione, 4th DIMENSION ignorerà il comando e non lo eseguirà, per evitare eventuali conflitti di scrittura.

La realizzazione di una base dati destinata all'impiego in multi-utenza presuppone la ricerca dei casi in cui potete riscrivere una scheda e la verifica di questa possibilità.

Facciamo un esempio:
- l'utente A modifica la scheda ROSSI dell'archivio [Clienti]
- l'utente B aggiunge una fattura nell'archivio [Fatture] sotto il nome di SIG.ROSSI.

Da un punto di vista logico, l'utente B può confermare la fattura soltanto se gli è possibile aggiornare il fatturato del SIG.ROSSI. Di conseguenza, dovrà avere accesso alla modifica della scheda ed aspettare che questa venga liberata dall'utente A.

Dovete effettuare questa verifica allo scopo di conservare la coerenza dei dati, altrimenti risulterà una differenza tra la somma dei totali delle fatture ed il fatturato del SIG.ROSSI. In tal caso, l'integrità relazionale dei dati tra l'archivio [Fatture] e l'archivio [Clienti] andrà persa.

5. Tipo di accesso alle schede

E' necessario verificare se una scheda può essere riscritta prima di modificarla o cancellarla.

La scelta del tipo di accesso ad una scheda avviene tramite i comandi **SOLO LETTURA** e **LETTURA SCRITTURA**, che si applicano ad un archivio.

Applicate **SOLO LETTURA** ad un archivio per segnalare al programma che volete caricare le schede di questo archivio per la **consultazione**. Ciò implica che non potrete modificare né cancellare queste schede. Servitevi di questo tipo di accesso per consultare, visualizzare, stampare, e così via.

Applicate **LETTURA SCRITTURA** ad un archivio per segnalare al programma che volete caricare le schede di questo archivio per la **modifica**. Un caricamento di questo tipo non vi garantisce la possibilità di riscrivere la scheda.

Durante il caricamento della scheda, si possono verificare due casi:
1. Un altro utente sta modificando la scheda; anche se chiedete al programma di caricarla per la modifica, la scheda vi verrà ritornata soltanto per la consultazione. Diciamo allora che la scheda vi è stata restituita **in uso**. In questo modo, 4th DIMENSION evita i conflitti di accesso ad una scheda ed impedisce qualsiasi modifica in contemporanea.
2. La scheda non è in modifica da parte di un altro utente: in tal caso, si dice che la scheda vi è stata restituita **non in uso**. Potete quindi modificarla o cancellarla.

Quando caricate una scheda per la consultazione, questa vi viene sempre restituita "in uso".

Quando caricate una scheda per la modifica, questa diventa **in uso su tutta la rete**. Gli altri utenti potranno caricare questa scheda soltanto per la consultazione. Se uno di loro tenta di caricarla per la modifica, il programma glie la restituirà per la consultazione.

Potete avere in uso una sola scheda per ogni archivio: e cioè la **scheda corrente**.

Per sapere se la scheda corrente è in uso, utilizzate la funzione **Scheda è in uso**.
Questa funzione restituirà **Vero** se abbiamo chiesto di caricare la scheda per la consultazione o se abbiamo tentato di caricarla per la modifica, ma senza successo.
Restituirà **Falso** se la scheda è stata caricata per la modifica.

Potete avere accesso ai valori contenuti dalla scheda indipendentemente dal fatto che la scheda sia stata caricata per la consultazione o per la modifica. A seconda del contesto, potrete invece modificarla o no, oppure cancellarla.

Quando caricate una scheda per la modifica, gli altri utenti possono accedervi solo più per la consultazione. Dopo che la scheda è stata modificata e registrata, occorre **sbloccarla sulla rete**, per consentire nuovamente agli altri utenti di accedervi per la modifica. Questa operazione è detta **liberare la scheda**.

4th DIMENSION libera la scheda quando cambiate scheda corrente. Se non siete sicuri di poterla cambiare subito, potete liberarla voi stessi chiamando il comando **LIBERA SCHEDA**, che libera sulla rete la scheda corrente dell'archivio specificata e la scarica dalla memoria del Macintosh.

In mono-utenza, 4th DIMENSION conserva una memoria tampone interna per ridurre il numero di accessi al disco. Lo scopo di questa zona temporanea è di mantenere nella memoria centrale le schede a cui accedete con maggior frequenza. Quindi, il primo caricamento di una scheda può non provocare la lettura sul disco, se questa è già stata inserita in questa zona di memoria.

Quando invece il programma viene usato in multi-utenza, non viene conservata alcuna memoria tampone interna. Quando caricate una scheda, questa viene sempre letta da disco.
In questo modo tutti i comandi, come RELAZIONE A UNO, che cambiano la scheda corrente di un archivio, leggono sempre la scheda da disco. Effettuano il cosiddetto **primo caricamento della scheda**, che avviene sempre a partire dal disco per garantirvi di avere sempre la versione aggiornata della scheda, nel caso in cui quest'ultima fosse stata modificata.

Dopo aver caricato una scheda, questa può essere disponibile soltanto per la consultazione, e per ottenerla per la modifica dovrete cercare di caricarla nuovamente chiamando il comando **CARICA SCHEDA**.

Questo comando non ha alcun effetto in mono-utenza, poiché la scheda corrente di un archivio è sempre in memoria. Al contrario, il funzionamento di CARICA SCHEDA è completamente diverso in multi-utenza. CARICA SCHEDA legge sempre la scheda su disco, che questa si trovi in memoria centrale o meno, e vi restituisce la scheda per la modifica, se lo avete specificato nel tipo di accesso all'archivio.

Il metodo per ottenere una scheda per la modifica può essere riassunto in questo modo:

- indicate che volete accedere all'archivio per la modifica, chiamando LETTURA SCRITTURA
- chiamate uno dei comandi che carica la scheda per la prima volta, ad esempio CERCA
- usate la funzione Scheda è in uso per sapere se la scheda è disponibile per la modifica
- altrimenti, ripetete la chiamata di CARICA SCHEDA finché Scheda è in uso vi restituirà Falso
- modificate la scheda, la registrate e poi la liberate cambiando scheda corrente o chiamando LIBERA SCHEDA.

Nota
Se cancellate una scheda, non è necessario chiamare LIBERA SCHEDA, perché questa non esiste più.

Quando aprite una base dati in multi-utenza, l'accesso al suo archivio è automaticamente predisposto per la modifica. Quando caricate una scheda per la consultazione, il programma non verifica se questa è in uso oppure se è libera. Scegliere l'accesso per la consultazione non è dunque un obbligo, ma vi permette di ridurre il volume del traffico sulla rete, ottimizzando così la velocità di accesso alla base dati.

6. I semafori

Abbiamo visto come accedere alle schede per la consultazione o per la modifica.

Il programma è in grado di gestire qualsiasi conflitto di accesso, e fa sì che due utenti non possano modificare la stessa scheda contemporaneamente. Si tratta della cosiddetta **integrità fisica** dei dati. Per mezzo dei comandi esaminati, conserviamo la cosiddetta **integrità logica** dei dati.

L'impiego di una base dati in multi-utenza può introdurre altre limitazioni. Consideriamo l'esempio di una base dati che integri una gestione contabile. Non vogliamo che due utenti possano effettuare contemporaneamente la chiusura di un periodo. Chiamiamo *FaiChiusura* la procedura incaricata di effettuare una chiusura e prima verifichiamo se esiste già una chiusura in corso.

Da un punto di vista pratico, occorre che ogni utente possa accedere ad una **zona comune sulla rete** che gli consenta:
- di assicurarsi che non esista nessuna chiusura in corso
- di segnalare agli altri utenti che ne sta iniziando una.

4th DIMENSION vi mette a disposizione un'area comune, chiamata **tabella dei semafori**, che è registrata nel **documento dei semafori**. La tabella può contenere fino a 1000 semafori, ognuno dei quali presenta un nome composto al massimo di 8 caratteri.

Potete accedere alla tabella dei semafori servendovi dei comandi **Semaforo** ed **ELIMINA SEMAFORO**.

Un semaforo può essere semplicemente paragonato ad una bandiera. Nel nostro esempio, vogliamo che la procedura FaiChiusura verifichi l'esistenza del semaforo chiamato "CHIUSURA".

Se il semaforo esiste, *FaiChiusura* non esegue alcuna azione, altrimenti lo crea nella tabella dei semafori, poi esegue la chiusura. Al termine dell'operazione, *FaiChiusura* cancella il semaforo dalla tabella per indicare che la chiusura non è più in corso.

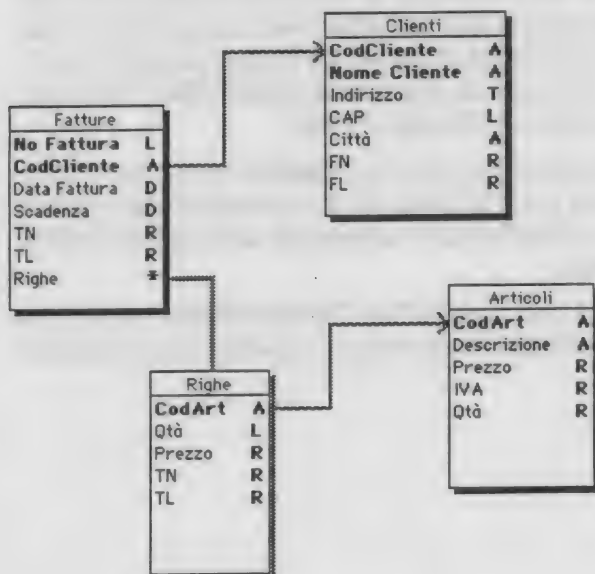
Eseguiamo il test e la creazione del semaforo servendoci della funzione **Semaforo**, che agisce nel modo seguente:
- la chiamiamo passando come parametro il nome del semaforo da verificare, in questo caso "CHIUSURA",
- la funzione esamina la tabella dei semafori per verificare se il semaforo esiste,
- in questo caso, la funzione restituirà **Vero**, altrimenti creerà il semaforo e successivamente restituirà **Falso**.

Notate che la funzione **Semaforo** non è "inattiva": non si limita a segnalarci se esiste un semaforo. Se non c'è, lo crea subito. Con il suo funzionamento, Semaforo **ci garantisce che due utenti non possono creare contemporaneamente lo stesso semaforo**.

Di conseguenza, la procedura *FaiChiusura* si presenta in questo modo:

Mentre(**Semaforo**("CHIUSURA"))	Verifica il semaforo "CHIUSURA"
MESSAGGIO("Attendi, è in corso una chiusura")	Se esiste, visualizza un messaggio per segnalarlo
Fine mentre	Continua il ciclo finché Semaforo restituisce Falso
	Quando il semaforo è stato creato, esegui
FaiChiusura	*FaiChiusura*
ELIMINA SEMAFORO("CHIUSURA")	Cancella il semaforo dalla tabella

7. Le transazioni

Abbiamo visto come possiamo impedire operazioni contemporanee usando i semafori. Esistono ancora situazioni in cui gli strumenti della multi-utenza già esaminati non ci permettono di gestire i conflitti in modo adeguato.
Esaminiamo la base dati la cui struttura è rappresentata in questa figura:

L'aggiunta di una fattura comporta da un lato l'aggiornamento del fatturato del cliente e dall'altro quello delle quantità vendute degli articoli indicati in fattura.

Esaminiamo le diverse tappe di quest'operazione:
- aggiungiamo una fattura al cliente ROSSI, costituita da una trentina di righe
- aggiorniamo la scheda dell'archivio [Clienti]. Per farlo, carichiamo questa scheda per la modifica. Eventualmente, aspettiamo che la scheda si liberi, la modifichiamo e la registriamo.
- procediamo perciò all'aggiornamento degli articoli indicati nella fattura. A questo scopo, carichiamo la prima scheda dell'archivio [Articoli] in modifica. Eventualmente, aspettiamo che la scheda si liberi, la modifichiamo e la registriamo. Ripetiamo quest'operazione per tutti gli articoli citati fattura.
- al termine, confermiamo l'inserimento della fattura.

Durante l'operazione, può succedere che un'altro utente cancelli un articolo indicato in fattura prima che abbiamo potuto procedere all'aggiornamento. Ci troviamo perciò nell'impossibilità di confermare una fattura che utilizzerebbe un prodotto diventato inesistente. Possiamo allora decidere di eliminare la riga corrispondente nella fattura, ma è logico confermare una fattura che non corrisponde alla realtà? Decidiamo quindi di non confermare la fattura. Ma attenzione! Dovremo tornare indietro. Occorre infatti annullare gli aggiornamenti già effettuati nell'archivio [Clienti] e nell'archivio [Articoli].

Conclusione: questa operazione è lunga, fastidiosa e infruttuosa. Uno strumento che ci consenta di confermare l'operazione nell'insieme o di annullarla se una delle schede è inaccessibile ci semplificherà il compito. 4th DIMENSION ci fornisce questo strumento: si tratta delle **transazioni**.

Per gestire una transazione avete a disposizione tre comandi: INIZIA TRANSAZIONE, CONFERMA TRANSAZIONE e ANNULLA TRANSAZIONE.

Aprite una transazione chiamando INIZIA TRANSAZIONE. Il programma vi riserva la base dati completa. Gli altri utenti si vedono impedire l'accesso alla base dati, sia per la modifica che per la consultazione.

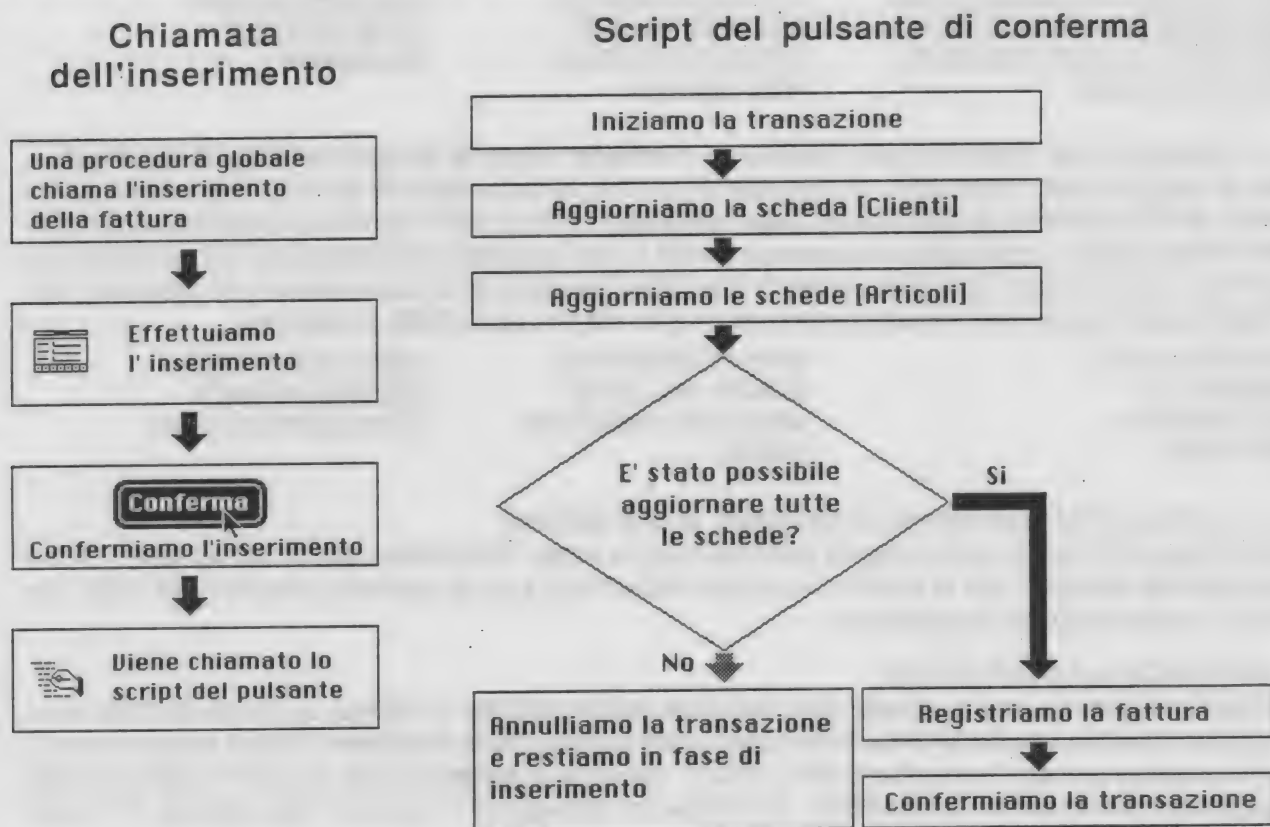
Dopo aver aperto una transazione, tutte le nuove scritture di schede non saranno più registrate direttamente nella base dati, ma in una zona temporanea della memoria centrale (o in un documento, se la memoria è satura).

Durante la transazione, verificate se le schede che volete riscrivere sono in uso e conservate una variabile booleana inizializzata a Vero che cambierete a Falso se una delle schede vi viene segnalata in uso. Al termine dell'operazione, questa variabile booleana vi permetterà di sapere se è stato possibile modificare tutte le schede. In tal caso, potrete confermare la transazione chiamando CONFERMA TRANSAZIONE, altrimenti annullerete l'operazione chiamando ANNULLA TRANSAZIONE.

Quando confermate una transazione, 4th DIMENSION riprende la zona temporanea in cui sono accumulate tutte le scritture delle schede, aggiorna la base dati e poi la libera. Gli altri utenti possono nuovamente accedervi. Quando annullate una transazione, 4th DIMENSION abbandona la zona temporanea e poi libera l'accesso alla base dati, che rimane così invariata.

Nello stesso istante può esistere una sola transazione. Se chiamate INIZIA TRANSAZIONE mentre una transazione è già stata iniziata da un altro utente, 4th DIMENSION aspetta che questa finisca, prima di iniziare la vostra.

Durante una transazione, lo stato della base dati su disco rimane invariato. Le schede che erano in uso presso altri utenti all'inizio della transazione rimangono tali. Una transazione non vi dispensa perciò dal verificare, all'atto del caricamento, se le schede che volete riscrivere sono in uso. Se vi viene ritornata in uso una scheda compresa nella transazione, destinata ad essere riscritta, dovrete annullare la transazione per evitare un ciclo senza fine. Infatti, se non la annullate, dovrete aspettare che venga liberata una scheda da un utente in attesa della fine della vostra transazione. In questo caso, ripetete la transazione.

Nel nostro esempio, l'uso di una transazione semplifica l'operazione di aggiunta di una fattura. Gestiamo la transazione nello script del pulsante di conferma:

Nota
Fate riferimento all'esempio completo descritto con il comando INIZIA TRANSAZIONE

---

4. I comandi di 4th DIMENSION e la multi-utenza

1. I comandi che caricano una scheda in memoria

La scelta del tipo di accesso all'archivio (consultazione o modifica) dipende dal contesto della chiamata del comando. Dipende dalla vostra necessità di avere o meno la possibilità di riscrivere ulteriormente la scheda corrente dell'archivio su cui agisce il comando.
E' comunque possibile mantenere l'accesso all'archivio per la modifica, in qualunque caso.

I vantaggi di poter verificare se è sufficiente un accesso per la consultazione, sono i seguenti:
- aumento della velocità di esecuzione: il programma non interrogherà il server per sapere se la scheda corrente ottenuta è in uso oppure no
- aumento della velocità di esecuzione: se la scheda non è in uso, il programma non tenterà di bloccarla inutilmente
- aumento della velocità di utilizzo: non bloccate inutilmente la scheda corrente, e di conseguenza la sua riscrittura da parte di un altro utente.

ULTIMA SCHEDA
VAI A SCHEDA
VAI A SCHEDA NELLA SELEZIONE
RELAZIONE A MOLTI PRECEDENTE
RELAZIONE A UNO PRECEDENTE
RELAZIONE A UNO
CERCA CON INDICE
CERCA
CERCA IN SELEZIONE
CERCA SU ESEMPIO
CERCA CON FORMULA
PRIMA SCHEDA
SCHEDA PRECEDENTE
SCHEDA SUCCESSIVA
RELAZIONE A MOLTI
TUTTE LE SCHEDE
USA INSIEME

2. I comandi che caricano una scheda in memoria senza lo scopo esplicito di riscriverla
Per la maggior parte delle volte, la chiamata di uno dei comandi elencati qui di seguito è a semplice scopo di consultazione. In altri termini, quasi certamente non si vorrà riscrivere la scheda corrente durante l'esecuzione o al termine del comando, perciò è utile impostare l'accesso per la consultazione per l'archivio su cui agisce il comando, al fine di ottimizzare l'impiego della base dati e di non bloccare inutilmente la scheda corrente, impedendo così ad un altro utente la possibilità di riscriverla.

ESPORTA TESTO
ESPORTA DIF
ESPORTA SYLK
RAPPORTO
GRAFICO SELEZIONE
STAMPA SELEZIONE
MAILING SU SELEZIONE
ORDINA
ORDINA CON FORMULA
ORDINA CON INDICE
VISUALIZZA SELEZIONE

3. I comandi che ammettono la riscrittura di una scheda
Questi comandi hanno, nella maggior parte dei casi, lo scopo di riscrivere schede già esistenti. Vi conviene quindi impostare per la modifica l'accesso dell'archivio a cui si riferisce il comando, se volete che questo venga eseguito normalmente.

3.1 APPLICA ALLA SELEZIONE
Questo comando riscrive le schede della selezione dell'archivio su cui agisce, se l'istruzione che viene applicata modifica il valore di almeno un campo della scheda. Se la selezione contiene schede in uso e se il comando provoca la riscrittura delle schede, troverete le schede in uso che non è stato possibile aggiornare nell'insieme **LockedSet**. E' sufficiente individuare il numero delle schede di questo insieme.
Se la chiamata di APPLICA ALLA SELEZIONE causa la riscrittura delle schede, vi conviene prima impostare l'accesso all'archivio in modifica. A questo scopo, applicate all'archivio il comando LETTURA SCRITTURA. Se per errore avete conservato il tipo di accesso per la consultazione, non verrà aggiornata alcuna scheda e ritroverete tutte le schede della selezione nell'insieme LockedSet al termine dell'esecuzione dell'istruzione.
Se la chiamata di APPLICA ALLA SELEZIONE non provoca la riscrittura delle schede, vi conviene prima impostare l'accesso all'archivio per la consultazione. A questo scopo, applicate all'archivio il comando SOLO LETTURA. Se per per errore avete conservato il tipo di accesso per la modifica, la procedura verrà eseguita normalmente, ma l'operazione non sarà ottimizzata: il programma cercherà inutilmente di caricare per la modifica e di bloccare ogni scheda della selezione.

Nota
Se avete dei comandi APPLICA ALLA SELEZIONE annidati che agiscono su archivi diversi, l'insieme LockedSet viene aggiornato al termine dell'esecuzione di ogni APPLICA ALLA SELEZIONE.

Esempi

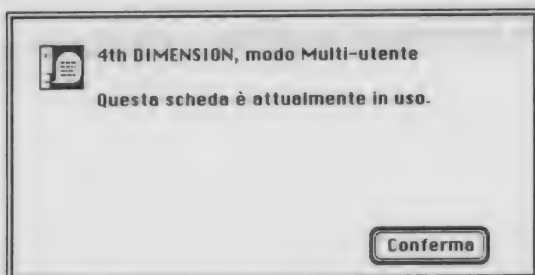
1. Esaminiamo il caso di un archivio [Fatture] con un campo [Fatture]Totale netto. Per una certa selezione dell'archivio, desideriamo calcolare la somma dei totali delle fatture al netto delle tasse. Non è necessario riscrivere le schede, la selezione in questo caso è richiamata soltanto per la consultazione. Possiamo scrivere una funzione simile a quella indicata qui di seguito:

```
IMPOSTA ARCHIVIO CORRENTE([Fatture])
SOLO LETTURA
LaSomma:=0
CERCA
Se(OK=1)&(Schede in selezione#0))
    APPLICA ALLA SELEZIONE(LaSomma:=LaSomma+[Fatture]Totale netto)
Fine se
$0:=LaSomma
```

2. Esaminiamo il caso di un archivio [Stipendi] con un campo[Stipendi]Base. Per una certa selezione dell'archivio, vogliamo applicare un aumento del 5% sulla base salariale. Occorre quindi riscrivere le schede della selezione. Possiamo scrivere una procedura di questo tipo:

```
IMPOSTA ARCHIVIO CORRENTE([Stipendi])
LETTURA SCRITTURA
CERCA
Se((OK=1))&(Schede in selezione#0))
NUOVO INSIEME("LockedSet")
Mentre(Schede in insieme("LockedSet")#0)
 APPLICA ALLA SELEZIONE([Stipendi]Base:=[Stipendi]Base*1,05)
 Se(Schede in insieme("LockedSet")#0)
    USA INSIEME("LockedSet")
 Fine se
Fine mentre
Fine se
```

3.2 MODIFICA SCHEDA

Questo comando riscrive la scheda corrente dell'archivio su cui agisce se confermate l'inserimento. Se non verificate se la scheda è in uso al momento della chiamata di MODIFICA SCHEDA e questa è effettivamente in uso, verrà visualizzato il seguente messaggio:

MODIFICA SCHEDA non verrà eseguito ed alla variabile di sistema OK assumerà il valore 0.

Esempi

1. Esaminiamo la seguente procedura:

IMPOSTA ARCHIVIO CORRENTE([Arch])
LETTURA SCRITTURA
CERCA
Se((OK=1)&(**Schede in selezione**#0))
 MODIFICA SCHEDA
Fine se

Questa procedura è corretta. Tuttavia, presenta due punti deboli:
a) Se durante la chiamata di **MODIFICA SCHEDA** la scheda è in uso, riceverete il messaggio di avviso e dovrete richiedere un'altra volta l'esecuzione della procedura per cercare di modificare la scheda.
b) Se avete potuto modificare la scheda perché non era in uso, la procedura non la libera. La scheda verrà quindi liberata soltanto quando, con le vostre operazioni, cambierete la scheda corrente dell'archivio: una scheda resta in uso inutilmente.

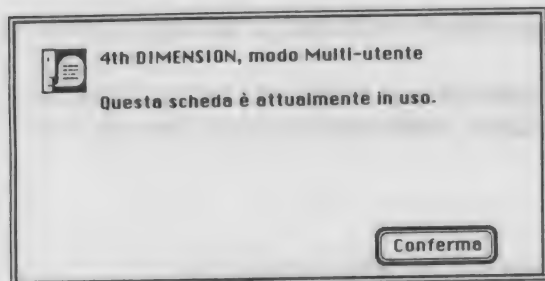
2. Vediamo come migliorare la procedura precedente:

IMPOSTA ARCHIVIO CORRENTE([Arch])
LETTURA SCRITTURA
CERCA
Se((OK=1)&(**Schede in selezione**#0))
 MODSCHEDA
Fine se

La procedura *MODSCHEDA* è la seguente:

$NumVolte:=0	Inizializza il contatore $NumVolte a 0
OK:=1	Inizializza la variabile a 1
Mentre(**Scheda è in uso**&(OK=1)	Mentre la scheda è in uso e OK=1
$NumVolte:=$NumVolte+1	Incrementa il contatore $NumVolte di un'unità
Se(**Modulo**($NumVolte;15))	Se il ciclo è stato eseguito per 15 volte
CONFERMA("Questa scheda è in uso presso un altro utente, vuoi attendere?")	Conferma o meno l'attesa
Fine se	
CARICA SCHEDA	Prova a caricare la scheda in modifica
Fine mentre	
Se(OK=1)	Se OK=1 significa che la scheda non è in uso
MODIFICA SCHEDA	Modifica la scheda
LIBERA SCHEDA	Libera la scheda
Fine se	

La procedura *MODSCHEDA* può essere usata come procedura generica per modificare la scheda corrente dell'archivio corrente.

3.3 MODIFICA SELEZIONE
Questo comando può riscrivere le schede su cui avete fatto doppio clic e di cui confermate le modifiche. Se la scheda su cui avete fatto doppio clic è in uso o se avete dimenticato di specificare l'accesso all'archivio per la modifica, verrà visualizzato il seguente messaggio:

L'unica precauzione che dovrete prendere quando chiamate il comando MODIFICA SELEZIONE è di applicare prima il comando LETTURA SCRITTURA all'archivio su cui state lavorando.

3.4 RICEVI SCHEDA, REGISTRA RELAZIONE PRECEDENTE
Se la scheda corrente su cui agisce uno di questi due comandi è in uso, non potrete in nessun caso registrare le modifiche apportate con questi comandi. Vi conviene quindi verificare prima se la scheda è in uso.

3.5 REGISTRA SCHEDA, REGISTRA RELAZIONE
Questi due comandi possono registrare una scheda appena creata o una già esistente su cui avete effettuato delle modifiche. Nel secondo caso, vi conviene verificare prima se la scheda è in uso. Nel primo caso, non si tratta di riscrittura ma di scrittura di una nuova scheda. L'utente che ha effettuato la creazione possiede la scheda per la modifica. Dovrà quindi liberarla per renderla disponibile per la modifica da parte degli altri utenti della base dati. Quest'ultima osservazione si riferisce anche al comando AGGIUNGI SCHEDA.

Nota
Non dimenticate che una modifica tramite formula del campo di una scheda, in particolare le modifiche delle sottoschede (AGGIUNGI SOTTOSCHEDA, CANCELLA SOTTOSCHEDA, e così via) presuppone implicitamente la riscrittura della scheda. Vi conviene perciò verificare prima se la scheda è in uso, e se si tratta di una scheda già esistente nella base dati.

3.6 CANCELLA SCHEDA
Vi consigliamo di verificare prima se la scheda è in uso. CANCELLA SCHEDA applicato ad una scheda in uso non ha alcun effetto.

3.7 CANCELLA SELEZIONE
Il funzionamento di questo comando è analogo a quello di APPLICA ALLA SELEZIONE. Ritroverete le schede che non è stato possibile cancellare nell'insieme LockedSet.

3.8 VETTORE IN SELEZIONE
L'accesso all'archivio su cui agisce il comando deve essere impostato prima per la modifica, se la selezione non è vuota.

4. I comandi specifici della multi-utenza

ANNULLA TRANSAZIONE
CARICA SCHEDA
INIZIA TRANSAZIONE
ELIMINA SEMAFORO
Scheda è in uso
LETTURA SCRITTURA
SOLO LETTURA
LIBERA SCHEDA
Semaforo
CONFERMA TRANSAZIONE

Questi comandi permettono di ottimizzare l'impiego di una base dati in multi-utenza ed assicurano la coerenza logica dei dati.
La presenza di questi comandi nelle procedure di una base dati usata in mono-utenza non influisce sull'esecuzione delle procedure. L'applicazione può essere usata indifferentemente in uno dei due modi:

Comando	*Effetto in mono-utenza*
ANNULLA TRANSAZIONE	si riferisce alla memoria tampone e non ad una transazione
CARICA SCHEDA	non ha effetto, poichè la scheda corrente è sempre in memoria
INIZIA TRANSAZIONE	si riferisce alla memoria tampone e non più ad una transazione
ELIMINA SEMAFORO	non ha effetto
Scheda è in uso	non ha effetto e restituisce sempre Falso
LETTURA SCRITTURA	non ha effetto
SOLO LETTURA	non ha effetto
LIBERA SCHEDA	non ha effetto
Semaforo	non ha effetto e restituisce sempre Falso
CONFERMA TRANSAZIOINE	si riferisce alla memoria tampone e non più ad una transazione.

5. I comandi che cambiano effetto in multi-utenza

5.1 MODIFICA PRIVILEGI, REGISTRA TAMPONE, ORDINA PERMANENTEMENTE
Questi tre comandi non hanno effetto in multi-utenza, ma la loro presenza all'interno di una procedura non interrompe l'esecuzione del seguito della procedura.

5.2 DISIMPILA SCHEDA, IMPILA SCHEDA
Questi due comandi funzionano normalmente in multi-utenza, ma non mantengono l'informazione di scheda in uso o meno.

6. I comandi che accedono ai documenti su disco

Aggiungi a documento
CARICA INSIEME
NUOVO DOCUMENTO
REGISTRA VARIABILI
ESPORTA TESTO
ESPORTA DIF
ESPORTA SYLK
IMPORTA TESTO
IMPORTA DIF
IMPORTA SYLK
CARICA VARIABILI
Apri documento
IMPOSTA COMUNICAZIONE
REGISTRA INSIEME
CANCELLA DOCUMENTO
USA FILTRO

Tutti questi comandi aprono, creano o cancellano un documento su disco. Non dimenticate che in multi-utenza chiunque intervenga sulla base dati mette in comune i suoi dati con gli altri utenti ed esegue le stesse procedure, perciò è possibile che due utenti cerchino nel medesimo istante di aprire, creare o eliminare un documento con lo stesso nome e nella stessa cartella. Vi consigliamo quindi di verificare gli errori e la variabile di sistema OK dopo ogni chiamata di uno di questi comandi.

5. I server di rete

4th DIMENSION funziona in multi-utenza con i server di rete che rispettano il protocollo AFP. Tale protocollo (Apple Filling Protocol), messo a punto dalla Apple Computer Inc., permette di condividere il disco rigido in lettura ed in scrittura, tramite la rete AppleTalk o una rete compatibile.
Al momento della stesura del presente manuale, i server di rete compatibili sono i seguenti:

AppleShare
- versione 2.0 o successive
- il server di rete è installato su un Macintosh
- il supporto fisico è del tipo LocalTalk (o compatibile) o EtherTalk

3+ Mac
- versione 2.01 o successive
- il server è un disco rigido 3Com (con server integrato) oppure installato su un IBM PC o compatibile
- il supporto fisico è LocalTalk (o compatibile) o EtherTalk

Tops
- versione 2.1 o successive
- il server è installato su un Macintosh o su un IBM PC o compatibile
- il supporto fisico è LocalTalk (o compatibile) o EtherTalk

AlisaShare
- versione 3.4 o successive
- il server è un VAX impiegato sotto VMS
- il supporto fisico è LocalTalk (o compatibile) o EtherTalk

PacerShare
- versione 5.2 o successive
- il server è un VAX impiegato sotto VMS
- il supporto fisico è LocalTalk (o compatibile) o EtherTalk

NetWare
- versione 2.15
- il server è un IBM PC o compatibile
- il supporto fisico è LocalTalk (o compatibile) o EtherTalk

Multi-utenza: comandi

SOLO LETTURA — READ ONLY

Sintassi

SOLO LETTURA({Archivio})

Parametri	Tipo	Descrizione
Archivio	Espr_Archivio	Archivio a cui si vuole accedere per la sola lettura

Descrizione

SOLO LETTURA cambia il tipo di accesso all'archivio designato da *Archivio* e lo imposta per la consultazione. Ogni caricamento di schede di questo archivio, successivo a questa chiamata, verrà effettuato per la consultazione. Non è possibile riscrivere una scheda caricata per la consultazione. Una chiamata di MODIFICA SCHEDA o di REGISTRA RELAZIONE, ad esempio, verrà ignorata.

Al contrario, questo tipo di accesso non ha alcuna influenza sulla creazione di nuove schede (AGGIUNGI SCHEDA, NUOVA SCHEDA, NUOVA SCHEDA SU RELAZIONE, ...)

Durante il caricamento di una scheda per la consultazione, 4th DIMENSION non interroga il server per sapere se la scheda è in uso. L'utilizzazione di SOLO LETTURA permette di risparmiare tempo: usandolo per caricare schede per la consultazione, eviterete di bloccarle inutilmente sulla rete.

SOLO LETTURA si riferisce soltanto ai caricamenti di schede successivi alla sua chiamata. Se la scheda corrente di *Archivio* viene caricata per la modifica, il tipo di accesso di tale scheda resterà invariato dopo la chiamata di SOLO LETTURA.

Se una scheda viene caricata per la consultazione, una chiamata di Scheda è in uso restituirà Vero.

Se il tipo di accesso ad un archivio è impostato per la consultazione, rimane tale finchè non lo modificate nuovamente, chiamando LETTURA SCRITTURA. L'accesso predeterminato per tutti gli archivi di una base dati aperta in multi-utenza è impostato per la modifica.

Quando chiamate uno dei comandi:

ESPORTA TESTO	GRAFICO SELEZIONE	SELEZIONE IN VETTORE
ESPORTA DIF	STAMPA ETICHETTE	VISUALIZZA SELEZIONE
ESPORTA SYLK	STAMPA SELEZIONE	
RAPPORTO	MAILING SU SELEZIONE	

4th DIMENSION passa automaticamente in consultazione per l'archivio interessato. Al termine dell'operazione, il programma ritorna al tipo di accesso precedente.

Compatibilità versione 1.0.x / versione 4.x.x
Se nelle realizzazioni già esistenti volete accedere agli archivi in modifica durante l'esecuzione di uno di questi comandi, dovrete cambiare il valore dell'intero n° 23 della risorsa CUST. Se l'intero è uguale a 0, avviene il passaggio automatico in consultazione, altrimenti il tipo di accesso all'archivio rimane invariato.

SOLO LETTURA non ha alcun effetto in mono-utenza.

Archivio è un argomento opzionale. Se questo argomento viene omesso, SOLO LETTURA si riferirà all'archivio corrente.

Esempio

Esaminiamo il caso di un archivio [Clienti] di cui vogliamo esportare la selezione chiamando INVIA SCHEDA per ogni scheda della selezione. L'operazione in questo caso è a solo scopo di consultazione, non modifichiamo, nè cancelliamo alcuna scheda. La scelta del tipo di accesso per la consultazione permette di ottimizzare, in termini di velocità di esecuzione, l'esportazione delle schede:

IMPOSTA ARCHIVIO CORRENTE([Clienti])	Seleziona l'archivio Clienti
SOLO LETTURA	Imposta il tipo di accesso all'archivio per consultazione
CERCA	Esegui una ricerca
Se((OK=1)&(**Schede in selezione**#0))	Se la ricerca viene confermata e se la selezione non è vuota
FaiEsport	Chiama la procedura che effettuerà l'esportazione
Fine se	

Riferimenti

LETTURA SCRITTURA, CARICA SCHEDA, Scheda è in uso

LETTURA SCRITTURA — READ WRITE

Sintassi

LETTURA SCRITTURA({Archivio})

Parametri	Tipo	Descrizione
Archivio	Espr_Archivio	Archivio a cui si vuole accedere per la modifica

Descrizione

LETTURA SCRITTURA cambia il modo di accesso all'archivio definito da *Archivio* e lo imposta per la modifica. Qualsiasi caricamento di schede in questo archivio, successivo a questa chiamata, verrà effettuato per la modifica.

Impostate il modo di accesso ad un archivio per la modifica quando volete modificare o cancellare delle schede di questo archivio, cioè riscriverle. Diversamente dal tipo di accesso per la consultazione che vi restituisce sempre delle schede in uso, quello per la modifica non vi garantisce che la scheda venga sempre caricata per la modifica: la scheda vi verrà restituita in uso se è già caricata per la modifica presso un altro utente. Dovrete quindi esplicitamente verificare la funzione Scheda è in uso per sapere se la scheda caricata è disponibile per la modifica oppure no, altrimenti rischiate che alcuni comandi, come REGISTRA SCHEDA o CANCELLA SCHEDA, vengano ignorati.

4th DIMENSION gestisce l'integrità fisica della base dati, sta invece a voi gestire quella logica delle informazioni in essa contenute. SOLO LETTURA è essenzialmente uno strumento di ottimizzazione del tempo rete consumato, mentre LETTURA SCRITTURA e Scheda è in uso sono strumenti che vi consentono di garantire l'integrità logica delle informazioni contenute nella base dati.

Esaminiamo l'esempio di una base dati contenente un archivio [Vendite] ed un archivio [Magazzino]. L'aggiunta di una scheda in [Vendite] causa l'aggiornamento di una scheda in [Magazzino]. Fra i due archivi esiste quindi una relazione di causa-effetto. In questi casi, dovete verificare la disponibilità della scheda di [Magazzino] per la modifica, altrimenti rischiate di introdurre un'incongruenza tra il magazzino e le vendite, ossia di perdere l'integrità logica dei dati.

LETTURA SCRITTURA si riferisce soltanto alle schede caricate dopo la chiamata di questo comando. Se la scheda corrente di *Archivio* è caricata per la consultazione, il suo tipo di accesso rimane invariato. Dovrete chiamare esplicitamente CARICA SCHEDA per ottenere l'accesso alla scheda per la modifica.

Quando il tipo di accesso ad un archivio è impostato per la modifica, rimane tale finchè non viene nuovamente cambiato dal comando SOLO LETTURA. Il tipo di accesso predeterminato per tutti gli archivi di una base dati aperta in multi-utenza è impostato per la modifica.

LETTURA SCRITTURA non ha ancun effetto in mono-utenza.

Archivio è un argomento opzionale. Se questo argomento viene omesso, LETTURA SCRITTURA si riferirà all'archivio corrente.

Esempio

Vogliamo modificare la selezione di un archivio [Persone]:

LETTURA SCRITTURA([Persone])	Richiedi l'accesso all'archivio [Persone] per la modifica
CERCA([Persone])	Cerca le schede da modificare
Se((OK=1)& **Schede in selezione**([Persone]))#0)	Se la ricerca è confermata e sono state trovate delle schede
MODIFICA SELEZIONE([Persone])	Chiama MODIFICA SELEZIONE
Fine se	Se non è stato chiamato LETTURA SCRITTURA, verrà ritornato il messaggio "Questa scheda è attualmente in uso", ogni volta che verrà fatto doppio clic su una scheda.

Riferimenti

SOLO LETTURA, Scheda è in uso, CARICA SCHEDA

Scheda è in uso — Locked

Sintassi

Scheda è in uso({Archivio})
➤ Booleano

Parametri	Tipo	Descrizione
Archivio	Espr_Archivio	Archivio in cui si vuole verificare se la scheda corrente è in uso

Descrizione

La funzione Scheda è in uso vi permette di sapere se la scheda corrente dell'archivio designato da *Archivio* è in uso oppure no. Vi permette inoltre di sapere se la scheda è stata caricata per la consultazione o per la modifica.

Scheda è in uso restituirà Vero se la scheda è solo in consultazione, in questo caso non potrete riscriverla. La funzione restituirà Falso se la scheda non è in uso, e in questo caso potrete modificarla o cancellarla.

Se la scheda è stata caricata mentre l' accesso all'archivio era impostato per la consultazione, Scheda è in uso vi restituirà sempre Vero. Se invece la scheda è stata caricata mentre l'accesso all'archivio era per la modifica, avremo due casi. Se, al momento del caricamento, la scheda è libera, vi verrà restituita per la modifica e Scheda è in uso restituirà Falso. Se la scheda è in modifica presso un altro utente, la scheda vi verrà restituita per la consultazione e Scheda è in uso restituirà Vero.

Se non controllate se una scheda è in uso e chiamate un comando di scrittura, come CANCELLA SCHEDA, il programma non lo eseguirà se la scheda è in uso.
4th DIMENSION vi garantisce l'integrità fisica della base dati, voi dovrete conservare l'integrità logica dei dati in essa contenuti servendovi di Scheda è in uso.

Dopo aver ottenuto una scheda per la modifica, vale a dire non in uso, non dimenticate, dopo averla riscritta, di chiamare LIBERA SCHEDA per liberarla nuovamente sulla rete e fare in modo che gli altri utenti vi possano accedere liberamente (a meno che, ovviamente non l'abbiate cancellata).

Scheda è in uso restituisce sempre Falso in mono-utenza.

Archivio è un argomento opzionale. Se questo argomento viene omesso, Scheda è in uso si riferirà all'archivio corrente.

Esempio

Scriviamo una funzione generica (che chiamiamo *TestInUso*) riferita all'archivio corrente, che ci permette di attendere la disponibilità di una scheda per la modifica o di interrompere l'operazione nel caso in cui non volessimo attendere. Se la scheda viene caricata per la modifica, la funzione restituirà Vero, altrimenti restituirà Falso:

$NumVolte:=0	Inizializza il contatore $NumVolte a 0
OK:=1	Inizializza la variabile OK a 1
Mentre(**Scheda è in uso**&(OK=1))	Finché la scheda è in uso e OK=1
$NumVolte:=$NumVolte+1	Incrementa il contatore $NumVolte di una unità
Se(**Modulo**($NumVolte;20))	Se il ciclo è stato eseguito per 20 volte
CONFERMA("Scheda occupata, vuoi attendere?)	Conferma o meno l'attesa
Fine se	
CARICA SCHEDA	Prova a caricare la scheda non in uso
Fine mentre	
$0:=(OK=1)	Restituisci il risultato di questa funzione booleana: $0:=(OK=1)

Una volta creata questa funzione, potete utilizzarla in una procedura di modifica della scheda:

IMPOSTA ARCHIVIO CORRENTE([Arch])	Seleziona l'archivio [Arch] come archivio corrente
LETTURA SCRITTURA	Chiedi di accedere alle schede per la modifica
CERCA	Scegli la scheda da modificare
Se((OK=1)&(**Schede in selezione**#0))	Se la ricerca viene confermata e la selezione non è vuota
Se(*TestInUso*)	Verifica se la scheda è libera oppure no
MODIFICA SCHEDA	Modifica la scheda
LIBERA SCHEDA	Libera la scheda
Fine se	
Fine se	

Riferimenti

LIBERA SCHEDA, CARICA SCHEDA, LETTURA SCRITTURA

CARICA SCHEDA — LOAD RECORD

Sintassi

CARICA SCHEDA({Archivio})

Parametri	Tipo	Descrizione
Archivio	Espr_Archivio	Archivio di cui si vuole caricare la scheda corrente

Descrizione

Con il tipo di accesso ad *Archivio* impostato per la modifica, se una scheda viene restituita solo per la consultazione, il solo mezzo per tentare di ottenerla per la modifica è quello di caricarla un'altra volta. Ma attenzione, un comando come RELAZIONE A UNO non ha alcun effetto se la scheda corrente è la stessa. Ecco perchè dovete usare CARICA SCHEDA che, diversamente dagli altri comandi, legge sempre la scheda da disco e cerca di ottenerla in modifica, se lo specificate nel tipo di accesso all'archivio.

CARICA SCHEDA non ha alcun effetto in mono-utenza.

Archivio è un argomento opzionale, se questo argomento viene omesso, CARICA SCHEDA si riferirà all'archivio corrente.

Esempio

Fate riferimento all'esempio descritto per Scheda è in uso.

Riferimenti

Scheda è in uso, LETTURA SCRITTURA

LIBERA SCHEDA — UNLOAD RECORD

Sintassi

LIBERA SCHEDA({Archivio})

Parametri	Tipo	Descrizione
Archivio	Espr_Archivio	Archivio di cui si vuole liberare la scheda corrente

Descrizione

LIBERA SCHEDA libera la scheda corrente dell'archivio indicato con **Archivio**. Se la scheda è stata caricata solo per la consultazione, LIBERA SCHEDA si limita a scaricarla dalla memoria. Se la scheda è stata caricata per la modifica, LIBERA SCHEDA la scarica dalla memoria rendendola libera sulla rete e disponibile per la modifica da parte degli altri utenti.

4th DIMENSION libera automaticamente la scheda corrente di un archivio non appena cambiate scheda corrente. LIBERA SCHEDA ha lo scopo di evitare di lasciare una scheda in uso sulla rete, qualora non aveste cambiato la scheda corrente dell'archivio.

Indipendentemente dal tipo di accesso ad un archivio, la conferma dell'aggiunta di una scheda la rende disponibile per la modifica, perchè è stata creata da voi e gli altri utenti vi possono accedere soltanto per la consultazione. Se volete renderla disponibile per la modifica da parte di altri utenti, dovrete sia cambiare scheda corrente, sia chiamare in modo esplicito LIBERA SCHEDA.

Dopo una chiamata di LIBERA SCHEDA, la scheda è sempre quella corrente dell'archivio, ma non si trova più in memoria. Per accedere ai valori dei campi, sarà necessario caricarla nuovamente in memoria usando CARICA SCHEDA.

LIBERA SCHEDA non ha alcun effetto in mono-utenza.

Archivio è un argomento opzionale, se questo argomento viene omesso, LIBERA SCHEDA si riferirà all'archivio corrente.

Esempio

Fate riferimento all'esempio descritto per Scheda è in uso

Riferimenti

Scheda è in uso, CARICA SCHEDA

Semaforo — Semaphore

Sintassi

Semaforo(IlSemaforo)
➤ Booleano

Parametri	Tipo	Descrizione
IlSemaforo	Espr_Alfa	Semaforo che si vuole controllare o creare

Descrizione

La funzione Semaforo permette di verificare l'esistenza del semaforo indicato con *IlSemaforo* e di crearlo, se non esiste. Se esiste, la funzione restituisce Vero, altrimenti lo crea e poi restituisce Falso.

Due utenti non possono creare lo stesso semaforo contemporaneamente. Se volete impedire a due utenti di eseguire contemporaneamente delle operazioni, è sufficiente verificare per ognuna di esse l'esistenza di un semaforo concordato ed eseguire l'operazione soltanto se quest'ultimo non esiste. Chiamate ELIMINA SEMAFORO per cancellare il semaforo da cui è dipesa l'esecuzione dell'operazione appena terminata.

La lunghezza del nome di un semaforo non può superare gli otto caratteri. Le maiuscole e le minuscole non sono distinte, mentre invece lo sono i caratteri diacritici (es: E=e, ma é≠ê).

E' possibile creare fino a 1000 semafori.
I semafori creati con la funzione Semaforo vengono introdotti nel documento dei semafori associato alla base dati. 4th DIMENSION utilizza questo stesso archivio per assegnare i numeri degli utenti che usano la base dati, quindi fate attenzione a non usare "1", "2",... come nome per i vostri semafori, per evitare eventuali conflitti con il programma.
Semaforo non ha effetto in mono-utenza e, in questo caso, restituisce sempre Falso.

Esempio

Vogliamo impedire che due utenti possano aggiungere schede nell'archivio [Fatture] nello stesso momento, perciò subordiniamo l'operazione alla creazione di un semaforo che chiameremo per convenzione "AggFatt".
Cominciamo a scrivere una funzione generica, che chiameremo *TestSemaforo*, che ci permette di attendere la disponibilità del semaforo il cui nome è stato passato come primo parametro, o di interrompere l'operazione qualora non volessimo attendere. Il secondo parametro passato alla funzione è il messaggio di attesa.

Funzione *TestSemaforo*

$NumVolte:=0	Inizializza il contatore $NumVolte a 0
OK:=1	Inizializza la variabile OK a 1
$Sem:=Vero	Inizializza la variabile $Sem a Vero
Mentre($Sem&(OK=1))	Finché il semaforo esiste e OK=1
$Sem:=**Semaforo**($1)	Verifica l'esistenza del semaforo
Se($Sem)	Se il semaforo esiste
$NumVolte:=$NumVolte+1	Incrementa il contatore $NumVolte di una unità
Se(**Modulo**($NumVolte;20)	Se il ciclo è stato eseguito per 20 volte
CONFERMA($2)	Conferma o no l'attesa
Fine se	
Fine se	Se esci dal ciclo con OK=1 significa che è stato possibile creare il semaforo
Fine mentre	
$0:=(OK=1)	Resituisci il risultato della funzione.

Per impedire l'inserimento contemporaneo di schede nell'archivio [Fatture], scriviamo una procedura simile a questa:

Se(*TestSemaforo*("AggFatt";"Un altro utente sta già aggiungendo fatture, vuoi attendere?"))	Verifica il semaforo
IMPOSTA ARCHIVIO CORRENTE([Fatture])	Se il semaforo è stato creato, autorizza l'operazione
FORMATO PAGINA("InserimentoFatt")	
OK:=1	
Mentre(OK=1)	
AGGIUNGI SCHEDA	Aggiungi le schede
Fine mentre	
ELIMINA SEMAFORO("AggFatt")	Libera il semaforo al termine dell'operazione.
Fine se	

Riferimenti
ELIMINA SEMAFORO

ELIMINA SEMAFORO — CLEAR SEMAPHORE

Sintassi

ELIMINA SEMAFORO(IlSemaforo)

Parametri	Tipo	Descrizione
IlSemaforo	Espr_Alfa	Semaforo che si vuole eliminare

Descrizione

ELIMINA SEMAFORO vi permette di liberare sulla rete il semaforo indicato con *IlSemaforo*.

Non dimenticate di cancellare un semaforo che avete creato, altrimenti resterà presente nel documento dei semafori della base dati.

Non cancellate mai un semaforo che non avete creato personalmente. Possono verificarsi due casi:
- se il semaforo è un numero di utente assegnato da 4th DIMENSION, potete provocare la distruzione parziale dei dati contenuti nella base dati
- se il semaforo è stato creato da un altro utente, annullate la logica di svolgimento delle operazioni da voi prevista.

Esempio

Fate riferimento all'esempio descritto per Semaforo

Riferimenti

Semaforo

INIZIA TRANSAZIONE — START TRANSACTION

Sintassi

INIZIA TRANSAZIONE

Descrizione

Chiamate INIZIA TRANSAZIONE per dare inizio ad una transazione. Dopo aver avviato una transazione, dovete terminarla confermandola con il comando CONFERMA TRANSAZIONE, oppure annullandola, con ANNULLA TRANSAZIONE.

Quando viene avviata una transazione, viene impedito l'accesso alla scrittura ed anche alla lettura a tutti gli altri utenti della base dati. Se la transazione viene confermata o annullata, l'accesso alla base dati è nuovamente autorizzato. Occorre quindi avviare una transazione per operazioni precise e di breve durata.

Quando una transazione è avviata, tutte le registrazioni delle schede vengono memorizzate in una zona temporanea (nella memoria centrale o in un documento temporaneo se la memoria è satura).

Se al termine dell'operazione confermate la transazione, 4th DIMENSION effettuerà l'aggiornamento della base dati in una sola volta, utilizzando le informazioni conservate nella zona temporanea. Se annullate la transazione, tutte le scritture di schede effettuate nel corso della transazione vengono ignorate e la base dati rimane invariata.

Può esistere una sola transazione in corso per ogni base dati. Se chiamate INIZIA TRANSAZIONE mentre una transazione è già stata avviata da un altro utente, 4th DIMENSION aspetterà la fine di questa transazione prima di iniziare quella richiesta.

Quando un utente avvia una transazione, il programma vieta a tutti gli altri di accedere alla base dati per la consultazione che per la modifica. Se uno di loro cerca di accedere alla base dati, 4th DIMENSION attenderà la fine della transazione in corso.

Se chiamate INIZIA TRANSAZIONE dopo avere già avviato una transazione, il comando non avrà alcun effetto.

Quando avviate una transazione, vi riservate l'accesso alla base dati, tuttavia questa operazione non influisce assolutamente sulle schede già in uso. Una transazione perciò non vi esonera dal verificare, all'atto del caricamento, se le schede che volete riscrivere sono in uso. Se una scheda compresa nella transazione e destinata ad essere riscritta vi viene ritornata per la sola consultazione, dovete annullare la transazione per evitare di generare un ciclo senza fine, nel quale dovreste aspettare la liberazione di una scheda da parte di un'utente in attesa della fine della vostra transazione. In tal caso, ripetete nuovamente la transazione.

Può essere impossibile aggiornare una scheda se si verifica un errore del genere: spazio su disco esaurito, presenza di un blocco danneggiato su disco, e così via. In tal caso, 4th DIMENSION genera un errore ed annulla automaticamente la transazione.
Se volete intervenire personalmente, vi consigliamo di installare una procedura di interruzione sugli errori, usando CHIAMA SU ERRORE prima di iniziare la transazione. All'interno di questa procedura, assegnate un valore ad una variabile che vi permetterà di sapere se si è verificato un errore, e di annullare perciò la transazione. Non dimenticate di disattivare la procedura di interruzione dopo la transazione.

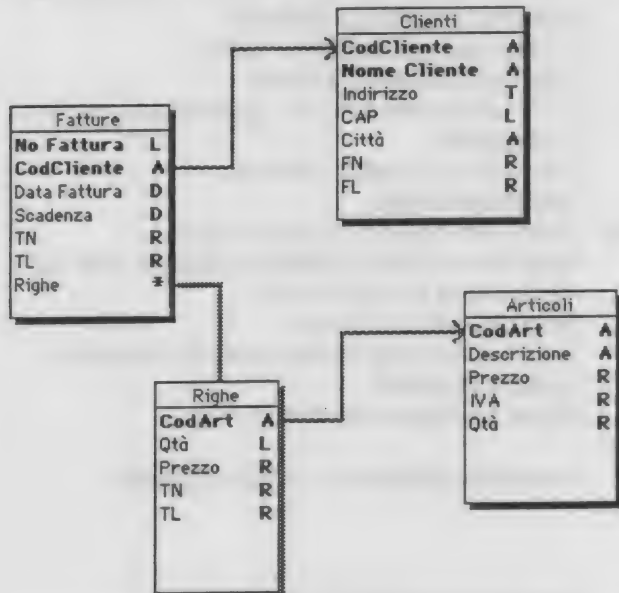
Una transazione vi permette di garantire l'integrità relazionale dei dati: infatti potete confermare un gruppo di schede con una sola azione o decidere di annullare l'intera operazione se non potete accedere per la modifica ad una della schede.

Nota
Quando viene avviata una transazione, 4th DIMENSION memorizza le registrazioni delle schede nella zona temporanea. Gli altri tipi di scrittura (INVIA PACCHETTO su una porta seriale o in un documento, REGISTRA VARIABILI, esportazione di schede, e così via) vengono eseguiti immediatamente. Occorre perciò effettuare queste operazioni soltanto dopo la transazione, e previa conferma della stessa.

Il funzionamento di una transazione in mono-utenza è analogo, con la sola differenza che non dovete preoccuparvi se le schede sono in uso.

Nota
Se chiamate INIZIA TRANSAZIONE in mono-utenza, 4th DIMENSION scrive su disco la memoria tampone interna prima di iniziare la transazione.

Esempio

Esaminiamo il caso di una base dati la cui struttura è rappresentata in questa figura:

Quando aggiungiamo una fattura, dobbiamo incrementare sia il fatturato del relativo cliente, sia le quantità vendute dei prodotti indicati nelle righe della fattura. A questo scopo, utilizziamo una transazione. Se è possibile aggiornare tutte le schede, confermiamo la transazione, altrimenti la annulliamo.

Scriviamo una procedura globale di aggiunta fatture, chiamata a partire da una scelta di menu:

IMPOSTA ARCHIVIO CORRENTE([Fatture])	Seleziona l'archivio [Fatture]
FORMATO PAGINA("InserFatture")	Seleziona il formato usato per l'inserimento
SOLO LETTURA([Clienti])	Scegli il tipo di accesso agli archivi [Clienti] e [Articoli] per
SOLO LETTURA([Prodotti])	la consultazione
Ripeti	
AGGIUNGI SCHEDA	Aggiungi le schede
Finché(OK=0)	

Nel formato "InserFatture", creeremo un pulsante, **Conferma**, che è un semplice pulsante senza alcuna azione associata. Gestiremo la transazione nello script del pulsante, indicato qui di seguito:

CHIAMA SU ERRORE("Precauzione")	Installa la procedura di interruzione su errore
TuttoOK:=Vero	Inizializza la variabile TuttoOK a Vero
INIZIA TRANSAZIONE	Avvia la transazione
LETTURA SCRITTURA([Clienti])	Chiedi l'accesso per la modifica sull'archivio
LETTURA SCRITTURA([Articoli])	[Clienti] e sull'archivio [Articoli]
CARICA SCHEDA([Clienti])	Carica la scheda del cliente *(Nota 1)*
TuttoOK:=**Non**(**Scheda è in uso**([Clienti]))	Verifica se la scheda è in uso
Se(TuttoOK)	Se la scheda non è in uso, aggiornala e continua la transazione
[Clienti]FN:=[Clienti]FN+[Fatture]TN	
[Clienti]FL:=[Clienti]FL+[Fatture]TL	Incrementa il fatturato del cliente
REGISTRA RELAZIONE(CodCliente)	Registra la scheda
Mentre(**Non**(**Fine sotto_selezione**(Righe))&TuttoOK)	Finché non hai agito su tutte le righe
RELAZIONE A UNO(Righe'CodArt)	Richiama la scheda dell'articolo indicato nella riga
TuttoOK:=**Non**(**Scheda è in uso**([Articoli]))	Verifica se la scheda è in uso
Se(TuttoOK)	Se la scheda non è in uso
[Articoli]Qtà:=[Articoli]Qtà+Righe'Qtà	Incrementa la quantità delle vendite dell'articolo
REGISTRA RELAZIONE(Righe'CodArt)	Registra la scheda
SOTTOSCHEDA SUCCESSIVA(Righe)	Passa alla riga successiva
altrimenti	
AVVISO("Inserimento annullato: articolo in uso")	La scheda dell'articolo è in uso, segnalalo
Fine se	
Fine mentre	
altrimenti	
AVVISO("Inserimento annullato: cliente in uso")	La scheda del cliente è in uso, segnalalo
Fine se	
Se(TuttoOK)	L'operazione si è svolta correttamente
ANNULLA	Non confermare l'inserimento della fattura *(Nota 2)*
REGISTRA SCHEDA	
CONFERMA TRANSAZIONE	Conferma la transazione
altrimenti	Se qualcosa ha impedito la conferma
ANNULLA TRANSAZIONE	Annulla la transazione *(Nota 3)*
Fine se	
SOLO LETTURA([Clienti])	Scegli nuovamente l'accesso per la consultazione per gli archivi [Clienti] ed [Articoli]
SOLO LETTURA([Articoli])	
LIBERA SCHEDA([Clienti])	Libera la scheda [Clienti] e l'ultima scheda [Articoli] caricata *(Nota 4)*
LIBERA SCHEDA([Articoli])	
CHIAMA SU ERRORE("")	Disattiva la procedura di interruzione su errore

Nota1

CARICA SCHEDA([Clienti])

1. La scheda corrente dell'archivio [Clienti] è quella selezionata sia con una relazione automatica sia con un'istruzione come RELAZIONE A UNO, cerchiamo quindi di caricarla in modifica.
2. Chiamiamo in questo caso CARICA SCHEDA per essere sicuri di leggere la scheda da disco e per avviare il test che verifica se la scheda è in uso.

Nota2

ANNULLA
REGISTRA SCHEDA
CONFERMA TRANSAZIONE

Registriamo personalmente la scheda per far sì che la registrazione appartenga alla transazione.
Se avessimo chiamato CONFERMA, la registrazione della fattura non avrebbe fatto parte della transazione.

Nota3

ANNULLA TRANSAZIONE
Se non è stato possibile aggiornare anche solo una scheda o se si è verificato un errore, annulliamo la transazione. 4th DIMENSION abbandona tutte le operazioni registrate nella zona temporanea e la base dati su disco rimane invariata. Restiamo invece in inserimento e potremo riprovare in seguito a confermare la fattura.

Nota4

LIBERA SCHEDA([Clienti])
LIBERA SCHEDA([Articoli])
La funzione di LIBERA SCHEDA in questo caso è doppia: da un lato, la scheda viene effettivamente liberata sulla rete, dall'altro, se non è possibile confermare la transazione sin dalla prima volta, occorre scaricare le schede in memoria per non incrementarle erroneamente al prossimo tentativo di conferma della transazione.

Procedura Precauzione:

TuttoOK:=**Falso**
AVVISO("Errore di accesso alla base dati. L'inserimento non può essere confermato.").

Riferimenti

CONFERMA TRANSAZIONE, ANNULLA TRANSAZIONE, CHIAMA SU ERRORE

CONFERMA TRANSAZIONE — VALIDATE TRANSACTION

Sintassi

CONFERMA TRANSAZIONE

Descrizione

CONFERMA TRANSAZIONE vi permette di confermare una transazione che avete avviato in precedenza con il comando INIZIA TRANSAZIONE. 4th DIMENSION riprende tutte le registrazioni delle schede memorizzate nella zona temporanea durante la transazione e le trasferisce nella base dati su disco. Se la base dati viene usata in multi-utenza, il programma libererà successivamente l'accesso alla base dati per tutti gli utenti.

Se non c'è nessuna transazione in corso, CONFERMA TRANSAZIONE non esegue alcuna azione.

Esempio

Fate riferimento all'esempio descritto per INIZIA TRANSAZIONE

Riferimenti

INIZIA TRANSAZIONE, ANNULLA TRANSAZIONE

ANNULLA TRANSAZIONE — CANCEL TRANSACTION

Sintassi

ANNULLA TRANSAZIONE

Descrizione

ANNULLA TRANSAZIONE vi permette di annullare una transazione avviata in precedenza con il comando INIZIA TRANSAZIONE. 4th DIMENSION abbandona tutte le registrazioni delle schede memorizzate nella zona temporanea durante la transazione e lascia la base dati su disco invariata. Se la base dati viene usata in multi-utenza, il programma libererà successivamente l'accesso alla base dati per tutti gli utenti.

Se non esiste nessuna transazione in corso, ANNULLA TRANSAZIONE non esegue alcuna azione.

Se durante una transazione create o aggiungete una o più schede ad un archivio, e se annullate la transazione, la selezione dell'archivio si riduce alla selezione vuota. L'ultima scheda creata rimane come scheda corrente; sarà quindi presente soltanto in memoria e non sarà compresa nella selezione.

Esempio

Fate riferimento all'esempio descritto per INIZIA TRANSAZIONE

Riferimenti

INIZIA TRANSAZIONE, CONFERMA TRANSAZIONE

4th DIMENSION

4 . LINGUAGGIO

ACI

Procedura speciale: Startup

In 4th DIMENSION, potete creare una procedura speciale, *Startup*, che verrà chiamata automaticamente da 4th DIMENSION all'apertura della base dati.
Vi permette di preparare la base dati inizializzando diverse variabili o vettori, di presentare una finestra di benvenuto all'utente o qualsiasi altra operazione di vostra scelta.

Esiste una limitazione nell'uso durante l'esecuzione della procedura *Startup*: qualsiasi operazione sui menu deve iniziare con IMPOSTA BARRA MENU(1).

Impostazioni particolari

La risorsa CUST ID=0

4th DIMENSION presenta due risorse che permettono di definire i parametri di funzionamento del programma.
La prima si chiama 'CUST' ed è contrassegnata con il numero 0.
Potete modificarla in qualsiasi momento direttamente nel programma servendovi di una procedura esterna.

N° dell'intero	Valore predeterminato	Significato
0	0	0 --> Non è possibile usare gli accessori della scrivania in fase di stampa 1 --> Gli accessori possono essere usati in fase di stampa
1	0	0 -->Non è possibile usare gli accessori della scrivania durante la visualizzazione di un indicatore di progressione 1 --> Gli accessori della scrivania sono disponibili durante la visualizzazione di un indicatore di progressione
2	0	Questo intero imposta la velocità di rotazione del cursore rotante 0 --> Nessuna visualizzazione del cursore rotante 16 -->Velocità media di rotazione
3	3	Codice ASCII del carattere che genera una nuova sotto-scheda durante l'inserimento. Predeterminato: Tabulatore.

4	256	Tasto modificatore associato al tasto precedente. Predeterminato: Comando. Valore da introdurre 256 --> tasto Comando 512 --> tasto Maiuscole 1024 --> tasto Blocco Maiuscole 2048 --> tasto Opzione
5	3	Codice ASCII del carattere che genera la conferma di una scheda in inserimento. Predeterminato: tasto Inserimento
6	0	Tasto modificatore associato al tasto precedente. Predeterminato: nessuno
7	46	Codice ASCII del carattere che genera l'annullamento di una scheda. Predeterminato: tasto "."
8	256	Tasto modificatore associato al tasto precedente. Predeterminato: tasto Comando
9	3	Codice ASCII del carattere che genera la conferma di una finestra di dialogo. Predeterminato: tasto Inserimento
10	0	Tasto modificatore associato al tasto precedente. Predeterminato: nessuno
11	46	Codice ASCII del carattere che genera l'annullamento di una finestra di dialogo. Predeterminato: tasto "."
12	256	Tasto modificatore associato al tasto precedente. Predeterminato: tasto Comando.
13	1	Informazioni relative alla finestra 0 --> La finestra ha le dimensioni dello schermo 1 --> La finestra occupa tutto lo schermo con la barra del titolo nascosta sotto la barra di menu 2 --> La finestra ha le dimensioni indicate dagli interi 14,15,16,17 3 --> La finestra ha le dimensioni indicate dagli interi 14,15,16,17 ed è centrata sullo schermo
14	-1	Lato superiore della finestra
15	-1	Lato sinistro della finestra
16	-1	Lato inferiore della finestra
17	-1	Lato destro della finestra
18	0	Cambiamento delle dimensioni della finestra nel passaggio dagli ambienti Progetto e Utilizzo all'ambiente Personalizzato 0 --> Le dimensioni della finestra non variano 1 --> Le dimensioni della finestra variano in base al valore contenuto nell'intero n°13
19	0	Cambiamento delle dimensioni della finestra dopo l'esecuzione di una procedura 0 --> Le dimensioni della finestra non variano 1 --> Le dimensioni della finestra variano in base al valore contenuto nell'intero n° 13
20	0	Riservato al programma; non utilizzare
21	0	Riservato al programma; non utilizzare
22	0	Riservato al programma; non utilizzare

23	0	Riservato al programma; non utilizzare
24	0	Riservato al programma; non utilizzare
25	0	Questo intero vi permette di specificare se il **Durante** delle procedure di formato viene chiamato quando l'utente non conferma l'inserimento, cioè quando fa clic su un pulsante di annullamento. Predeterminato: questa fase viene chiamata. Se assegnate il valore 1 a questo intero, **Durante** non verrà eseguito per gli annullamenti.
26	0	Riservato al programma; non utilizzare
27	0	Confronti diacritici di stringhe di caratteri 0 --> Confronto non diacritico di stringhe di caratteri 128 --> Confronto diacritico di stringhe di caratteri 129 --> Usato per Paesi come la RFT o la Svezia
28	0	Impostazione automatica della sola lettura per gli archivi 0--> 4th DIMENSION passa automaticamente in consultazione soltanto per i comandi elencati nella sezione relativa alla multiutenza 1--> Nessun passaggio automatico in consultazione.

La risorsa Cach ID=512

La seconda risorsa dei parametri di 4th DIMENSION viene chiamata Cach. Vi permette di scegliere i parametri per l'occupazione di memoria di 4th DIMENSION e non può essere modificata durante l'uso di 4th DIMENSION.

Num Intero Lungo	**Valore predeterminato**	**Significato**
0	614400	Spazio di memoria riservato all'applicazione Predeterminato: 600 Kb
1	50	Percentuale della memoria restante riservata alla memoria tampone di 4th DIMENSION Predeterminato: 50%
2	22	Percentuale minima di memoria tampone trasferita su disco Predeterminato: 22%
3	3	Permette di definire il numero di master pointer riservati all'avvio di 4th DIMENSION. Questo numero viene definito dividendo il numero dei blocchi da 64 byte della memoria tampone per il valore di questo intero. Modificatelo solo se siete veramente esperti.

Codici di controllo

Messaggi all'apertura di una base dati

Non posso aprire il driver di stampa.
Non sarà possibile stampare o mostrare l'unità le unità di misura nei formati.
State usando 4th DIMENSION con una Cartella Sistema che non contiene il driver di stampa selezionato con la **Scelta Risorse** del menu **Mela**.

La base dati non può essere aperta in quanto è stata creata con una versione di 4th DIMENSION più recente.
State cercando di aprire una base dati di 4th DIMENSION creata o modificata con una versione del programma più recente di quella che possedete.

Memoria insufficiente per avviare il programma.
Non avete memoria sufficiente per usare 4th DIMENSION con una memoria insufficiente. 4th DIMENSION necessita di almeno 700 Kb di memoria centrale. Verificate la memoria tampone del vostro Macintosh scegliendo **Pannello di Controllo** nel menu Mela.

Inserisci il disco Utilità
4th DIMENSION vi richiede il disco originale del programma: inserite il disco **Utilità** nell'unità disco interna.

E' avvenuta un'interruzione registrando l'archivio: Nome dell'archivio
L'ultima volta in cui avete lavorato con la base dati si è verificato un problema durante la registrazione di una scheda. Probabilmente la scheda non è stata registrata correttamente. Si tratta di una **scheda non valida**. Riparate la base dati con 4D TOOLS.

E' avvenuta un'interruzione cancellando una scheda dall'archivio: Nome dell'archivio
L'ultima volta in cui avete lavorato con la basedati si è verificato un problema al momento della cancellazione di una scheda. Probabilmente la scheda non è stata cancellata correttamente. Si tratta di una **scheda danneggiata**. Riparate la base dati con 4D TOOLS.

Usa i programmi di utilità per riparare la base dati.
All'apertura di una base dati, se è visualizzato uno dei messaggi precedenti, 4th DIMENSION vi indica che probabilmente è necessario riparare la **base dati** con il programma 4D TOOLS.

E' avvenuta un'interruzione registrando le risorse.
Durante l'ultima utilizzazione della base dati si è verificato un errore passando dall'ambiente **Struttura** all'ambiente **Utilizzo** o **Personalizzato**. Il problema è avvenuto durante la registrazione di un formato, di una procedura di formato o di archivio, o di una procedura globale.

Non posso caricare la base dati. Opera su una copia effettuata in precedenza.
Quando aprite una base dati, se si presenta questo messaggio, 4th DIMENSION vi indica che la struttura della base dati è probabilmente danneggiata. L'unica soluzione che avete è di ripartire dalla registrazione precedente della base dati.

Messaggi sulle parole d'ordine

Questa parola d'ordine non permette di usare questo formato.
Siete in ambiente **Utilizzo** o **Personalizzato** e state cercando di usare un formato protetto da una parola d'ordine che non è la vostra.

Questa parola d'ordine non permette di usare questo menu.
Siete in ambiente **Personalizzato** e state tentando di usare un menu protetto da una parola d'ordine diversa dalla vostra.

Messaggi generali

Questo campo non può essere utilizzato per una ricerca.
Avete scelto **Cerca**... nel menu **Selezione**. State cercando, nella finestra standard per la ricerca, di effettuare una ricerca su un campo di tipo **Immagine**. Non è possibile effettuare una ricerca su un campo di tipo **Immagine**.

Non posso registrare la ricerca
Avete scelto **Cerca**... o **Cerca con una Formula**... nel menu **Selezione** e avete fatto clic su **Registra**. 4th DIMENSION non può registrare la ricerca: lo spazio su disco è insufficiente, sono stati raggiunti i limiti della directory, e così via.

Non posso registrare il rapporto
Avete scelto **Stampe** nel menu **Rapporti** e **Registra** nel menu **Archivio**. 4th DIMENSION non può registrare il rapporto: lo spazio su disco è insufficiente, sono stati raggiunti i limiti della directory, e così via.

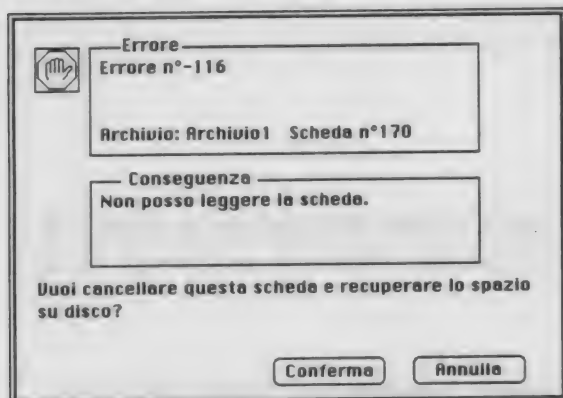
Non posso registrare il filtro ASCII
Avete scelto uno dei due comandi per la creazione di un filtro ASCII nel menu **Varie** e avete fatto clic su **Registra**. 4th DIMENSION non può registrare il filtro: lo spazio su disco è insufficiente, sono stati raggiunti i limiti della directory, e così via.

Non ci sono dati indicizzati
Avete scelto **Cerca** con **Indice...** nel menu **Selezione** su un archivio che non possiede campi indicizzati. Questo stesso messaggio può comparire anche in ambiente Personalizzato se utilizzate, all'interno di una procedura, la finestra standard per la ricerca con indice.

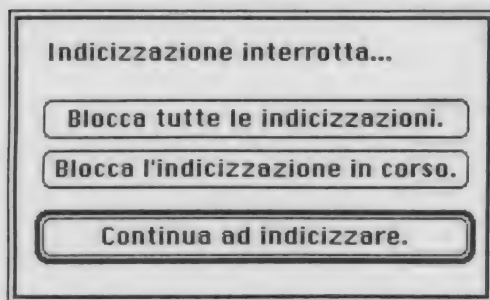
Non ho trovato nessuna scheda.
In seguito ad una ricerca con indice non è stata trovata nessuna scheda. Questo stesso messaggio può comparire anche in ambiente Personalizzato se utilizzate, all'interno di una procedura, la finestra standard per la ricerca con indice.

Non c'è spazio per ordinare i dati
Lo spazio in memoria centrale (o lo spazio su disco) non è sufficiente per ordinare la selezione delle schede. Ci sono troppe schede oppure troppi livelli di ordinamento.

Errore di lettura di una scheda

Se durante la lettura di una scheda 4th DIMENSION rileva un errore, vi verrà presentato un messaggio che indica il numero dell'archivio e quello della scheda in questione.

Potrete scegliere di cancellare questa scheda o tentare di recuperarla usando 4D TOOLS.

Interruzione di una indicizzazione

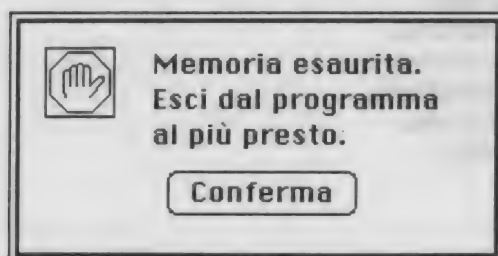
Se avete interrotto una indicizzazione, verrà visualizzata la seguente finestra di dialogo:

Avete a disposizione tre possibilità:

Blocca tutte le indicizzazioni
Questa opzione interrompe tutte le indicizzazioni e cancella tutti gli indici non costruiti della struttura

Blocca l'indicizzazione in corso
Questa opzione interrompe l'indicizzazione in corso sul campo e cancella l'indice anche dalla struttura

Continua ad indicizzare
Questa opzione vi permette di continuare l'indicizzazione come se nulla fosse successo.

Memoria esaurita

La finestra seguente appare se 4th DIMENSION non ha memoria sufficiente per effettuare l'operazione richiesta. Questo messaggio può comparire in qualsiasi momento durante l'uso dalla base dati. Ritornate allora al Finder ed aumentate la quantità di memoria assegnata a 4th DIMENSION.

Codici di errore della base dati

Errore n°4001	Numero di archivio non valido.
Errore n°4002	Numero di scheda non valido.
Errore n°4003	Numero di campo non valido.
Errore n°4004	Non esiste alcuna scheda o sotto-scheda corrente
Errore n°-9998	Chiave già esistente.
Errore n°-9999	Sono stati raggiunti i limiti di spazio del documento .data.
Errore n°-9997	E' stato raggiunto il numero massimo di schede.
Errore n°-9996	La pila è piena.
Errore n°-9995	Limite della Versione Dimostrativa.
Errore n°-9994	Comunicazione interrotta dall'utente.
Errore n°-9993	Barra di menu danneggiata.
Errore n°-9992	Parola d'ordine già esistente.
Errore n°-9991	Non hai l'autorizzazione all'accesso.
Errore n°-9990	Il Timeout è stato raggiunto.
Errore n°-9989	Struttura non valida.
Errore n°-10000	Indirizzo del dato non valido.
Errore n°-10001	Struttura dell'indice non valida.
Errore n°-10002	Struttura della scheda non valida.
Errore n°-10003	Numero di scheda non valido.
Errore n°-10004	Numero pagina dell'indice non valido.

Codici di errore del linguaggio di 4th DIMENSION

Errore n°1	Era attesa una "(".
Errore n°2	Era atteso il nome di un campo.
Errore n°3	Questo comando può essere eseguito solo su un campo di un sotto archivio.
Errore n°4	I parametri della lista devono essere tutti dello stesso tipo.
Errore n°5	Non so a quale archivio applicare questo comando.
Errore n°6	Questo comando si può applicare solo su un campo di tipo sotto-archivio.
Errore n°7	Era atteso un valore numerico.
Errore n°8	Era atteso un valore alfanumerico.
Errore n°9	Era atteso un risultato di un test condizionale.
Errore n°10	Questo comando non si può applicare a questo tipo di dato.
Errore n°11	Questo comando non si può applicare tra due test condizionali.
Errore n°12	Questo comando non si può applicare tra due argomenti numerici.
Errore n°13	Questo comando non si può applicare tra due valori alfanumerici.
Errore n°14	Questo comando non si può applicare a due valori di tipo data.
Errore n°15	Questa operazione non si può applicare a questi due valori.
Errore n°16	Questo campo non è in relazione.
Errore n°17	Era atteso il nome di un archivio.
Errore n°18	I tipi non sono compatibili.
Errore n°19	Questo campo non è indicizzato.
Errore n°20	Era atteso un "=".

Errore n°21	Questa procedura non esiste.
Errore n°22	I campi devono appartenere allo stesso archivio o sotto archivio per ordinare o per creare un grafico.
Errore n°23	Era atteso un "<" o un ">".
Errore n°24	Era atteso un ";".
Errore n°25	Troppi campi per effettuare un ordinamento.
Errore n°26	Il campo deve essere di tipo alfa, data o numerico.
Errore n°27	Il campo deve essere preceduto dal nome dell'archivio di appartenenza.
Errore n°28	Il campo deve essere di tipo numerico.
Errore n°29	Il valore deve essere 1 o 0.
Errore n°30	Era attesa una variabile.
Errore n°31	Non esiste una barra di menu con questo numreo.
Errore n°32	Era attesa una data.
Errore n°33	Funzione o comando non implementato.
Errore n°35	Gli insiemi appartengono ad archivi diversi.
Errore n°36	L'archivio è danneggiato.
Errore n°37	Era atteso un ":"
Errore n°39	L'insieme non esiste.
Errore n°40	E' una funzione, non una procedura.
Errore n°41	Era attesa una variabile o un campo di un sotto archivio.
Errore n°42	Non posso togliere la scheda dalla pila.
Errore n°43	Non posso trovare la funzione.
Errore n°44	Non posso trovare la procedura.
Errore n°45	Era atteso un cmapo o una variabile.
Errore n°46	Era atteso un algoritmo numerico o alfanumerico.
Errore n°47	Il campo deve essere di tipo alfanumerico.
Errore n°48	Errore di sintassi.
Errore n°49	Qui non può essere usato questo operatore.
Errore n°50	Questi operatori non possono essere usati insieme.
Errore n°51	Modulo non implementato.
Errore n°52	Era atteso un vettore. / Timeout sulla porta seriale.
Errore n°53	L'indice del vettore è fuori dai limiti consentiti.
Errore n°54	I valori non sono compatibili.
Errore n°55	Era atteso un valore di tipo booleano.
Errore n°56	Era atteso un campo, una variabile o un archivio.
Errore n°57	Era atteso un operatore.
Errore n°58	Era attesa una ")".
Errore n°59	Non era atteso questo tipo di argomento.
Errore n°1006	Procedura interrotta dall'utente.

Codici di errore numerici indicati da SANE

±INF Questo codice in realtà non segnala un errore, ma viene indicato quando effettuate calcoli del tipo: R/0 dove R#0 e R#∞, oppure R±∞ con R≠±∞.

NaN(37) Potenza x^y non valida. Si tratta di una potenza x^y non valida. Nel seguente esempio, questo errore si verifica sulla radice quadrata, poichè abbiamo utilizzato la seguente istruzione: vSottoarchivio:= -5^(1/2).

Non si può ottenere un numero reale che, moltiplicato per se stesso, dia un valore negativo.

NaN(002) Addizione non valida. Si verifica sommando ∞ con -∞. In effetti, non è possibile sommare un infinito ad un infinito.

NaN(004) Divisione non valida. Si verifica ad esempio dividendo zero per zero.

NaN(008) Moltiplicazione nulla. Si verifica ad esempio moltiplicando zero per infinito.

NaN(017) Errore di conversione di una stringa in valore numerico. Si verifica durante la conversione di particolari stringhe in valore numerico. Quando 4th DIMENSION interpreta un valore numerico partendo da un valore alfanumerico, accetta i caratteri **"0"**, **"1"**, ... **"9"**, **","**, **"+"**, **"-"** **e** **"e"**, ossia le dieci cifre arabe, la virgola come segno di separazione decimale ed il simbolo "e" che separa la mantissa dall'esponente dei numeri espressi in notazione scientifica.
Gli altri caratteri vengono ignorati.

Esempi
Num("a1b0d0") restituisce 100 (i caratteri a, b e d vengono ignorati)
Num("15e2") restituisce 1500 (in effetti, 15e2 significa 15*10^2)
Num("Dicembre 1988") restituisce NaN(017), perchè SANE non può interpretare ee 1988.

NaN(036) Logaritmo non valido. Si verifica durante il calcolo di un logaritmo non valido, ossia di qualsiasi numero minore di 0. Ad esempio, Log(-1). Notate che Log(0) restituisce -INF.

Codici di errore restituiti dal sistema operativo

a/ Accesso al disco

Errore n°-33	Directory su disco piena.
Errore n°-34	Disco pieno.
Errore n°-35	Archivio non trovato.
Errore n°-36	Errore di input/output.
Errore n°-37	Nome dell'archivio o del volume errato.
Errore n°-38	L'archivio non è aperto.
Errore n°-39	Tentativo di lettura oltre la fine dell'archivio.
Errore n°-40	Tentativo di lettura prima dell'inizio dell'archivio.
Errore n°-41	Non posso aprire il documento.
Errore n°-42	Troppi archivi aperti.
Errore n°-43	Archivio non trovato.
Errore n°-44	Disco protetto da scrittura.
Errore n°-45	Archivio protetto da scrittura.
Errore n°-46	Disco protetto da scrittura.
Errore n°-47	Archivio in uso.
Errore n°-48	Esiste già un archivio con il nome ed il numero di versione specificati.
Errore n°-49	Archivio già aperto.
Errore n°-53	Il volume non è in linea.
Errore n°-54	Tentativo di scrittura su un archivio protetto.
Errore n°-61	I privilegi di accesso non permettono la scrittura.

b/ Memoria

Errore n°-108 Spazio in memoria insufficiente

Codici di errore del Resource Manager Macintosh

Errore n°-192 Risorsa non trovata.

Errore n°-193 Documento di risorse non trovato.

Tabella dei codici ASCII

Tabella ASCII

Car.	Dec.	Ott.	Esad.	Car.	Dec.	Ott.	Esad.	Car.	Dec.	Ott.	Esad.	Car.	Dec.	Ott.	Esad.
NUL	0	0	0	SP	32	40	20	@	64	100	40	`	96	140	60
SOH	1	1	1	!	33	41	21	A	65	101	41	a	97	141	61
STX	2	2	2	"	34	42	22	B	66	102	42	b	98	142	[illegible]
ETX	3	3	3	#	35	43	23	C	67	103	43	c	99	143	63
EOT	4	4	4	$	36	44	24	D	68	104	44	d	100	144	64
ENQ	5	5	5	%	37	45	25	E	69	105	45	e	101	145	65
ACK	6	6	6	&	38	46	26	F	70	106	46	f	102	146	66
BEL	7	7	7	'	39	47	27	G	71	107	47	g	103	147	67
BS	8	10	8	(	40	50	28	H	72	110	48	h	104	150	68
HT	9	11	9	)	41	51	29	I	73	111	49	i	105	151	69
LF	10	12	A	*	42	52	2A	J	74	112	4A	j	106	152	6A
VT	11	13	B	+	43	53	2B	K	75	113	4B	k	107	153	6B
FF	12	14	C	,	44	54	2C	L	76	114	4C	l	108	154	6C
CR	13	15	D	-	45	55	2D	M	77	115	4D	m	109	155	6D
SO	14	16	E	.	46	56	2E	N	78	116	4E	n	110	156	6E
SI	15	17	F	/	47	57	2F	O	79	117	4F	o	111	157	6F
DLE	16	20	10	0	48	60	30	P	80	120	50	p	112	160	70
DC1	17	21	11	1	49	61	31	Q	81	121	51	q	113	161	71
DC2	18	22	12	2	50	62	32	R	82	122	52	r	114	162	72
DC3	19	23	13	3	51	63	33	S	83	123	53	s	115	163	73
DC4	20	24	14	4	52	64	34	T	84	124	54	t	116	164	74
NAK	21	25	15	5	53	65	35	U	85	125	55	u	117	165	75
SYN	22	26	16	6	54	66	36	V	86	126	56	v	118	166	76
ETB	23	27	17	7	55	67	37	W	87	127	57	w	119	167	77
CAN	24	30	18	8	56	70	38	X	88	130	58	x	120	170	78
EM	25	31	19	9	57	71	39	Y	89	131	59	y	121	171	79
SUB	26	32	1A	:	58	72	3A	Z	90	132	5A	z	122	172	7A
ESC	27	33	1B	;	59	73	3B	[	91	133	5B	{	123	173	7B
FS	28	34	1C	<	60	74	3C	\	92	134	5C	|	124	174	7C
GS	29	35	1D	=	61	75	3D	]	93	135	5D	}	125	175	7D
RS	30	36	1E	>	62	76	3E	^	94	136	5E	~	126	176	7E
US	31	37	1F	?	63	77	3F	_	95	137	5F	Del	127	177	7F

Tabella ASCII estesa

Car.	Dec.	Ott.	Esad.	Car.	Dec.	Ott.	Esad.	Car.	Dec.	Ott.	Esad.	Car.	Dec.	Ott.	Esad.
Ä	128	200	80	†	160	240	A0	¿	192	300	C0	‡	224	340	E0
Å	129	201	81	°	161	241	A1	¡	193	301	C1	·	225	341	E1
Ç	130	202	82	¢	162	242	A2	¬	194	302	C2	‚	226	342	E2
É	131	203	83	£	163	243	A3	√	195	303	C3	„	227	343	E3
Ñ	132	204	84	§	164	244	A4	ƒ	196	304	C4	‰	228	344	E4
Ö	133	205	85	•	165	245	A5	≈	197	305	C5	Â	229	345	E5
Ü	134	206	86	¶	166	246	A6	∆	198	306	C6	Ê	230	346	E6
á	135	207	87	ß	167	247	A7	«	199	307	C7	Á	231	347	E7
à	136	210	88	®	168	250	A8	»	200	310	C8	Ë	232	350	E8
â	137	211	89	©	169	251	A9	…	201	311	C9	È	233	351	E9
ä	138	212	8A	™	170	252	AA		202	312	CA	Í	234	352	EA
ã	139	213	8B	´	171	253	AB	À	203	313	CB	Î	235	353	EB
å	140	214	8C	¨	172	254	AC	Ã	204	314	CC	Ï	236	354	EC
ç	141	215	8D	≠	173	255	AD	Õ	205	315	CD	Ì	237	355	ED
é	142	216	8E	Æ	174	256	AE	Œ	206	316	CE	Ó	238	356	EE
è	143	217	8F	Ø	175	257	AF	œ	207	317	CF	Ô	239	357	EF
ê	144	220	90	∞	176	260	B0	–	208	318	D0		240	360	F0
ë	145	221	91	±	177	261	B1	—	209	321	D1	Ò	241	361	F1
í	146	222	92	≤	178	262	B2	“	210	322	D2	Ú	242	362	F2
ì	147	223	93	≥	179	263	B3	”	211	323	D3	Û	243	363	F3
î	148	224	94	¥	180	264	B4	‘	212	324	D4	Ù	244	364	F4
ï	149	225	95	µ	181	265	B5	’	213	325	D5	ı	245	365	F5
ñ	150	226	96	∂	182	266	B6	÷	214	326	D6	ˆ	246	366	F6
ó	151	227	97	∑	183	267	B7	◊	215	327	D7	˜	247	367	F7
ò	152	230	98	∏	184	270	B8	ÿ	216	330	D8	¯	248	370	F8
ô	153	231	99	π	185	271	B9	Ÿ	217	331	D9	˘	249	371	F9
ö	154	232	9A	∫	186	272	BA	⁄	218	332	DA	˙	250	372	FA
õ	155	233	9B	ª	187	273	BB	¤	219	333	DB	˚	251	373	FB
ú	156	234	9C	º	188	274	BC	‹	220	334	DC	¸	252	374	FC
ù	157	235	9D	Ω	189	275	BD	›	221	335	DD	˝	253	375	FD
û	158	236	9E	æ	190	276	BE	ﬁ	222	336	DE	˛	254	376	FE
ü	159	237	9F	ø	191	277	BF	ﬂ	223	337	DF	ˇ	255	377	FF

I caratteri di controllo, che non possono essere stampati, sono indicati con la normale abbreviazione ASCII. Non hanno necessariamente un significato in ambiente Macintosh.
Diamo comunque i loro significati standard:

Codice		Nome	Significato	Implementazione su Macintosh
$00	00	NUL	nul character	
$01	01	SOH	Start Of Header	
$02	02	STX	Start Of Text	
$03	03	ETX	End Of Text	Tasto A-capo della tastiera o tasto Inserimento del tastierino numerico
$04	04	EOT	End Of Transmission	
$05	05	ENQ	Enquiry	
$06	06	ACK	Acknowledge	
$07	07	BEL	Bell	
$08	08	BS	BackSpace	Tasto di Ritorno Unitario della tastiera
$09	09	HT	Horizontal Tab	Tasto di Tabulazione della tastiera
$0A	10	LF	Line Feed	
$0B	11	VT	Vertical Tab	
$0C	12	FF	Form Feed	
$0D	13	CR	Carriage Return	Tasto A-capo
$0E	14	SO	Shift Out	
$0F	15	SI	Shift In	
$10	16	DLE	Data Link Escape	
$11	17	DC1	Device Control 1	
$12	18	DC2	Device Control 2	
$13	19	DC3	Device Control 3	
$14	20	DC4	Device Control 4	
$15	21	NAK	Negative Acknowledge	
$16	22	SYN	Synchronous Idle	
$17	23	ETB	End of Transmission Block	
$18	24	CAN	Cancel	
$19	25	EM	End Of Medium	
$1A	26	SUB	Substitute	
$1B	27	ESC	Escape	Tasto Esc della tastiera
$1C	28	FS	File Separator	Freccia a sinistra
$1D	29	GS	Group Separator	Freccia a destra
$1E	30	RS	Record Separator	Freccia in alto
$1F	31	US	Unit Separator	Freccia in basso
$7F	127	DEL	Delete	Tasto di Cancellazione del tastierino numerico.

Variabili di sistema e insiemi di sistema

Attenzione: Questo tipo di variabili e di insiemi gestiti da 4th DIMENSION non esistono all'apertura della base dati; vengono definiti dopo aver chiamato almeno una volta l'operazione che li aggiorna.

Variabili di sistema

OK
Questa variabile viene costantemente aggiornata da 4th DIMENSION. Vi permette di sapere se un'operazione è stata confermata o annullata.

Document
Questa variabile contiene il nome del documento creato in seguito alla chiamata di uno dei seguenti comandi, nel caso in cui questi visualizzino la finestra di dialogo per l'apertura o la creazione dei documenti: IMPOSTA COMUNICAZIONE, Nuovo documento, Apri documento, Aggiungi a documento, IMPORTA SYLK, ESPORTA SYLK, IMPORTA TESTO, ESPORTA TESTO, IMPORTA DIF, ESPORTA DIF, RAPPORTO, STAMPA ETICHETTE.

FldDelimit
Codice ASCII del separatore di campo durante un'importazione o esportazione di schede. FldDelimit ha il valore predeterminato di 9 (Tabulatore).

RecDelimit
Codice ASCII del separatore di schede durante un'importazione o esportazione di schede. RecDelimit ha il valore predeterminato di 13 (Ritorno carrello).

Error
Questa variabile contiene il numero dell'errore che ha provocato l'interruzione della procedura di 4th DIMENSION. E' accessibile soltanto in una procedura di interruzione su errore, installata con il comando CHIAMA SU ERRORE.

MouseDown
Questa variabile assume il valore 1 se il bottone del mouse viene premuto durante una chiamata su evento, ed è significativa soltanto in una procedura di interruzione su evento, installata con il comando CHIAMA SU EVENTO.

KeyCode
Questa variabile contiene il codice ASCII del tasto premuto durante una chiamata su evento ed è significativa soltanto in una procedura di interruzione su evento installata con il comando CHIAMA SU EVENTO.

Modifiers
Questa variabile contiene il valore dei modificatori (tasti Comando, Opzione, Control, Maiuscole, Blocco Maiuscole) durante una chiamata su evento ed è significativa soltanto nella procedura di interruzione su evento installata dal comando CHIAMA SU EVENTO.

Insiemi di sistema

UserSet
Insieme delle schede selezionate dall'utente durante la chiamata di VISUALIZZA SELEZIONE o MODIFICA SELEZIONE

LockedSet
Insieme delle schede in uso durante la chiamata dei comandi CANCELLA SELEZIONE o APPLICA ALLA SELEZIONE, in multi-utenza. Per informazioni più dettagliate, fate riferimento al capitolo relativo alla gestione in multi-utenza.